HUNTER NALLEY 99'ERS NEWS



TI 99/4A

MUEIRIRY CHIRISTIMAS



DISCLAIMER

The HUPP NEWS is the official newsletter of the HUNTER VALLEY NINETY NINE USER GROUP. Whilst every effort is made to ensure the correctness and accuracy of the information contained therin, be it of general, technical, or programming nature, no responsibility can be accepted by HV99 NEWS as a result of applying such information.

TEXAS INSTRUMENTS trademarks, names and logos are all copyright to TEXAS INSTRUMENTS.

HV99 is a non profit group of TI99/4A computer users, not affiliated in any way with TEXAS INSTRUMENTS.

CONTRIBUTIONS

Members and non members are invited to contribute articles for publication in HV99 NEWS.

Any copy intended for publication may be typed. hand written. or submited on tape/disc media as files suitable for use with TI Writer (ie. DIS/FIX 80 or DIS/VAR 80). A suitable Public Domain word processor program will be supplied if required by the club librarian Al Lawrence.

Please include along with your article sufficient information to enable the file to be read by the EDITOR eg. File Name etc.

The preferred format is **36** columns and page length 66 lines, right justified.

All articles printed in HV99 NEWS (unless notified otherwise) are considered to be PUBLIC DOMAIN. Other user groups wishing to reproduce material from HV99 NEWS may feel free to do so as long as the source and author are recognised.

Articles for publication can be submitted to.

THE EDITOR HV99 NEWS 15 GAYTON CLOSE WARNERS BAY 2282 NEWCASTLE

General address for ALL other club related correspondence.

THE SECRETARY HV99 USER GROUP 25 RESERVE RD. WANGI 2267 NEWCASTLE

YOUR COMMITTEE 1985

П.	WRIGHT	PRES.	PH. 458120
Ρ.	COXON	SECT.	PH. 751930
в.	RUTHERFORD	TRES.	PH. 458184
п.	LAMRENCE	LIBR.	PH. 486508
s.	TAHLOR	EDIT.	PH. 487076
B.	WOODS	EDAS.	PH. 662307
T.	MC. GOVERN	TECH.	PH. 523168
в.	MAC. CLURE	MODL.	PH. 437431
G.	JONES	TECH.	PH. 573744
D.	WINTON		PH. 591883

1986 R. G. M.

So ; Expa

STA

SPAC

dis

900

LOAI

more

thre

much

you

scre

STAC

SPAC

byte

Now

see'

Hmm

let:

BYTE

PROG

24 t

the

1,11

50

1,11

scre

2448

50

loca

I

Uhh?

WHAT

BYTE

GRAV

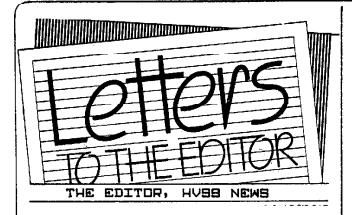
VIA

The Annual General Meeting for 1986 of the HUNTER VALLEY 99'ers USER GROUP will be held on 11th March 1986. All financial Members are requested to attend the Lake Macquarie Cultural Centre at 7:00 pm sharp on the above date. The Secretary will accept nomimations for any of the three Executive positions and six committee positions until 7:00 pm on 11/3/36

A11 financial members entitled to nominate for position the Executive on Committee. The current Committee would like to stress the need for people to nominate for for these positions. For the H.V.99'ers to continue to be a vibrant and active Club, new people and new ideas are required. No one person or one particular group has any right to claim that they or it are the keepers of all wisdom in thees matters. The flow of. initiatives and new Ideas can only serve to help to strengthen the Club over the next year.

So now is the time to assess your commitment to the Club and to the fundamentals of a User group.

If you do want to nominate, then please complete the enclosed nomination and return it to the Secretary no later than 7:00pm on 11/3/85. The forms must be signed by the person nominating and by a witness who must also be a financial Member of the H.V.99'ers.



** THE CASE OF THE MISSING BYTES **

So you just fitted the 32K Matchbox Expansion and after switching on you type >SIZE< and if all is well you are greeted with >13928 BYTES OF STACK FREE, 24488 BYTES OF PROGRAM SPACE FREE(you beaut! now I can disable that pesky quit key, so in >CALL INIT :: CALL LOAD(-31806,16)< great, it works! no accidental quitting halfway through a program, but I wonder how much space that used up? so again you type >SIZE< immediately the >13094 BYTES OF displays screen STACK FREE, 24488 BYTES OF PROGRAM SPACE FREE(, so it only used 24 bytes, very good.

Now down to some programming, see? >100 FOR I=1 TO 10 :: NEXT I<, Hmm I wonder how much that used? lets see >SIZE< whats this?? >13928 BYTES OF STACK FREE, 24468 BYTES OF PROGRAM SPACE FREE(I know !! those 24 bytes must of somehow been put in the program space, thats no good I'll start again and get rid of them so I'll type >NEW<, now lets see, I'll check >SIZE< Whats this ??, on screen >13928 BYTES OF STACK FREE, 24488 BYTES OF PROGRAM SPACE FREE somehow the disable memory location must have been cleared, so I will just quit and start again Uhh? the quit key wont function NOW THE HECK HAPPENED TO THOSE 24 WHAT BYTES ????.

R. KLEINSCHAFER HV99 GRAWIN VIA WALGETT 2832



ALEVE AND KEGKENC

Several years ago Texas Instruments (TI) developed computer called the 99/4A. Ιt didn't have all the bells and whistles o f something like Macintosh or IBM-AT, however, at one point you could buy fifty 99/4As for the same price as one Macintosh. Consequently, TI sold quite a few of them (over 2.5 million in the United States alone).

TI, as you probably know, is a very large, diversified corporation with many divisions and a wide range of products and services. Because of competition and continued loss of revenue, TI was forced in 1983, to abandon it's interests in the home computer market and act as though the 99/4A had never existed. This left the 99/4A without a parent company for support — an orphan computer. This is where my story begins!

This was such an abrupt occurence that no one had any idea what would happen to the 99/4A over 2 million users. its obvious possibilities were 99/4As would become dust collectors in users closets or flea specials for a dollar. However, because the machines were reliable and the market penetration widespread, the 50 survived its infancy and is coming of age in 1985.

Since TI did not encourage third party development of products for the 99/4A, the transition was not an easy one for its developers. Large companies like AtariSoft and Imagic continued to develop new

products for a short time, but of the smaller companies were able to survive the landslide effect σf unsold TI products being dumped at below cost prices.

As the dust settled over to 10 months a second generation of TI developer began to Corcomp gained recognition emerge. quality manufacturer a οf advanced expansion products for the 99/4A. Navarone Industries, who had weathered the storm with products like Console Writer, the Cartridge Expander and Disk Fixer is still a major force in the development of new products for the TI.

99/4A owners began developing a network of dedicated user groups, sharing ideas at monthly meetings and passing the word around that the TI was still alive and growing. The number of user groups grew to over 300 with some metropolitan clubs reporting memberships of over 2,000. As the word spread, international users were brought into the network.

With such a revival taking place in the TI community, against what would seem incredible odds, you would think that computer magazines would be having a hayday with news stories, editorials, product reviews, and advertising. However, every major publication that supported the 99/4A abandoned almost as abruptly as TI had done with little notification or reason. As new developers emerged so did new magazines. MICROpendium and 99 along with numerous Mini-Mag newsletters generated by local user Hunchback is a fairly hopeless game groups have become the communication and hasn't many good points medium for the 99/4A.

the today, and are continuing to grow. People are awakening to the fact you. that the \$US 50.00 computer they expandable and powerful than ever harder. At a point though the guard before. Not only have the users stays at the same speed and the game their efforts, developers have been working together to provide better, more compatible products for the 99/4A. Corcomp and Navarone have, for some time, been developing hardware and and it does get very monotonous. software products that make the best use of both product's capabilities.

Another exciting, and unprecedented event taking place in the TI community is а program development by Navarone whereby user groups can participate as a major contributor in the development of new products. This program allows TI users an opportunity to let developers know first hand, they want developed for their machine. They even get to see products before they are released offer and can suggestions enhancements and improved design. Some clubs have even initiated their own development projects that, when completed, will distribute finished product through Navarone.

The TI99/4A is definitely alive and kickin' and coming out of the There's a groundswell of closet. for the suport computer that promises to make it an exciting and productive tool for years to come. So, if you have one sitting in the closet you might strongly consider dusting it off and putting it to Just a modest work. investment will return you years of today enjoyment and satisfaction and it just may turn out to be the computer that you've always wanted.

Chuck Humphries (Corcomp Cursor)

JUNIUR SUFTWARE REVIEW

HUNCHBACK.

The graphics at the start of the game look promising, there is a The TI99/4A user base is one of hunchback on the top of a castle and largest and most dedicated of he tries to ring the bell before the their guard at the bottom of the castle wall climbs up the wall and shoots

The first time you do this it is a few years ago is more fairly easy but it gradually gets but some becomes very simple. After a while you become so sick of the game that you get yourself killed or quit the whole game.

Hunchback is a fairly boring game

Nick Coren HV99

He 19 ha un of pe HE Sc Ex r tł

te Ir **#** 1 A٤ 34

1 h f tl t

E P t

S N r 9 8

> 3 a

and ice in ogram user major nt of 110ws let what their new eased for sign. lheir when their

the ll of that and ome. the lider to

live

he.

ment of it uter

ame the

a

and the tle ots

is ets ard ame

ile hat the

ame

SECRETARY'S NOTES BY PETER COXON

Hello again for the last time in 1985!Well what a great meeting we had on the 12th November, with an unexpected visit fromBernie Elsner of TIUP W.A. I'm sure that those people who managed to get there were more than interested in the Software Demo's and the new Extended Basic Module from Germany. Many thanks to Bernie for taking the time to travel up from Sydney to pay um a flying vimit!

In the past few weeks we have had a flood of Newsletters from around Australia and Overseas. Ausie Post seems to be back on the right track again, so check out the Magazine Library and catch up on all the latest happenings! The news we all been waiting for has finally arrived from the U.K. on the workings of the Smart Cable, D. Hewitt of Oxford, thanks to England. We will be making a Prototype soon and let you all know the results!

If you receive your Stop Press!! Newsletter before 8th December, remember ' our Christmas get-together, departing the Community Centre 10.30a.m. at Sunday 8.12.85. We will travel by Bus to our destination, where there will be Swimming, Sporting Activities, bring your own BBQ Lunch and Drinks. A fun day for only \$1.00.(approx). per family. Why not join us, leave the keyboard at home and have some fun with the kids!

Finally I would like to thank you all for the support you have given in 1985. It has been a good year for the T.I.Users. and maybe 1986 will be even better!

Hope you enjoy your Bumper Issue. The Committee and I would like to wish you all a very happy christmas, and even happier 1986.

Christmas comes but once a year, and don't we think it's grand. Our money seems to go in one, and out the other hand.

If we could save the money, we spend on Christmas cheer, we'd have enough to last us throughout the coming year.

The T.I. is just one expense as dollars we do spend, for printers, Disks and Modules, it never seems to end. COMPUTER WIDOWS we've become, our men are all possesed, when will they program their machines to give us all a rest.

It could cook the tea, bath the kids, dry the washing when it rains! If their machines could manage this, then I would not complain.

The bugs and bytes they talk about mean nothing much to me. There's Basic and Extended, and Mini Memory. They spend their time with (calling sound) and also (calling clear), but when WE try to call them, somehow they never hear!

I can't begin to understand the hours that they keep, they read and load and edit when the household's fast asleep.

The games they play take over, and frustration settles in. they've got have another go they cannot let it win!

You know just what I'm saying, I bet you've been there too, but never mind its all for fun, I'm sure we'll see it through.

When I think about it, we should'nt make a fuss, because it really is a fact there's folks worse off than us!!!!

MERRY CHRISTMAS AND A HAPPY HEALTHY NEW YEAR.

Audrey Coxon HV99.



EL-PRESIDENTE'S CHRISTMAS MESSAGE



MUZIRIRY CHIRISTIMAS

The hand of fellowship and friendship which has been extended to the H.V.99'ers from within Australia and overseas is extreemly grafitying. We accept this in the spirit in which it has been extended.

My personal views ดก this matter are very basic. I firmly believe that people generally have a willingness to assist and Firstly however one has to indicate that the sharing and assistance will be appreciated and Once this has been established, goodwill is generally a matter of course. On the other hand, persons or groups that have inner cliches, are secretive and seem to have a policy of not giving people a "FAIR GO" are doomed to finally disappear into oblivion. HUNTER VALLEY 99'ers HAVE NO INTENTION OF DISAPPEARING ANYWHERE!!

I have often heard it said that the BEST way to get something done is to set about the task yourself. In the six months of our existance it can be fairly said that the H.V.99'ers. brought some truth to that statement.

The HUNTER VALLEY 99'ers is a fine monthly publication with a high content σf information. We are indebted to our Editor Steve Taylor interest, time and effort which he puts into it's publication. also want thank regular contributors to the News Letter. will strive to maintain that high level of local input. To achieve this end, each and every Member of the Group should look at ways in which a contribution can be made.

Amongst all the things which have so far been achieved. Two which stand out have been the production by Tony McGovern of Funnelwriter and the 32K project by Gary Jones.

Tony's Funnelwriter is a master stroke and for all those people who are using it I thank him.

Gary Jones with the tailoring

the Perth 32k MATCHBOX expansion into a kit project has assisted of our members to have access to some of the hidden power of T.I.99/4A. Also a thank you to those Members who gave their time to help other Members who assistance to install the expansion into their machines. This have been a project in the true USER GROUP spirit.

Still some more thanks are in order. Firstly Al Lawrence for excellent j ob does he on difficult task αf running our Software library. Ιt not uncommon to drive past Al's house after midnight and see the lights on in the T.I. Dungeon. Peter our Secretary whose fingers must be sore from all the letter writing upon him. Bob McClure our module librarian. Brain Wood who has hardly had time to take a breath since becoming publications our librarian. And Brian Rutherford our treasurer who keeps our books with the same accuracy with which he can hunt down bugs in programmes.

I have no intention of leting this become the largest thank you in recorded history. Those who have not been mention have no doubt been personally thanked by me for their efforts.

To the future, I am looking forward to the next year with delight. The Group I hope will

sti the alr

she thi que exp at is per

is per cor dat fol

pri sel of ack cre

tak Mem fro the Mer

Yea

fin

43

hieve er of Which Two the Ωf kt by а those him. bring asion 18 s to the - to e to eded sion been ROUP e in the the our not ouse s on our sore osed du l e has eath ions

king

with

NEWS ith a ginal la our the ch he also gular We high s in our with can ting u in have been heir

still further organise itself and high level of participation already in existance will increase even further.

There are two loose leaf sheets included in the News Letter this month. The first, is a questionnaire which i s explanatory and should be returned at the February meeting. The second the Club data sheet for each person, could you kindly make any corrections needed and also add any data missing. The formatt is as follows;

> TITLE SURNAME FIRST NAME **ADDRESS** TOWN STATE POST CODE CLUB POSITION NUMBER IN FAMILY (NAMES OPT.) INTERESTS IN DESCENDING ORD. YEAR JOINED M/SHIP No. EXTENDED BASIC SPEECH 32K EXP MINI MEM P.E. BOX RS232/PIO DISC DRIVE PRINTER

The programme listed below will print to screen the Calender for any selected month and year. The source of the programme is not know hence no acknowledgement is made to it's creator

To close off I would like to take this oportunity to wish all the Members of our Group, our friends from other Groups and the whole of the Home Computer Community a very Merry Christmas and an enjoyable New Year.

I will let Longfellow have the final say with this extract from "CHRISTMAS BELLS."

I heard the bells on Christmas Day Their old, familiar carols play, And wild, and sweet The words repeat Of peace on earth, good-will to all!.

> MERRY CHRISTMAS. Allen Wright.

110 !*HUNTER VALLEY 99'ERS* 120 !* 130 !* CALENDER PRINTER * 140 !* 150 ! ************ 160 CALL CLEAR 170 PRINT "ENTER MONTH AND Y EAR": :" eg. 02,1988" 180 INPUT J,I 190 IF J<=0 THEN 160 220 K=1 230 GOSUB 1000 240 N2=N 250 GOSUB 2010 260 N1=N :: J1=J 280 J=J+1 290 IF J>12 THEN N3=31 ELSE GOSUB 2010 :: N3=N-N1 340 NAMS=SEGS("JANFEBMARAPRM AYJUNJULAUGSEPOCTNOVDEC",3*J 1-2,3) 350 PRINT : : 360 PRINT TAB(9); NAM#; " "; I: 370 PRINT TAB(3); "S M Т F 5" T 380 L=0 385 FOR I=1 TO 6 390 FOR J=1 TO 7 400 IF (J(N2+1)AND(I=1) OR(L) =N3)THEN PRINT " ";ELSE L= L+1 :: PRINT SEG\$(" "&STR\$ "&STR#(L))-2,4); (L), LEN(" 430 NEXT J 440 PRINT 450 NEXT I 460 PRINT : : "TRY AGAIN Y OR N ? " 470 CALL KEY(0,K,S) 480 IF S<1 THEN 470 490 IF K=89 THEN 160 1000 IF J>2 THEN M8=J-2 :: Y 8=I ELSE M8=J+10 :: Y8=I-1 1070 C8=INT(Y8/100':: D8=Y8-100*C8 1090 N=INT((13*M8-1)/5)+K+D8 +INT(D8/4)+INT(C8/4)-C8-C8+7 1100 N=N-7*INT(N/7) 1110 RETURN 2010 N=INT(3055*(J+2)/100)-9 1 2020 GOSUB 3000 2030 IF J>2 THEN N=N-2+L 2070 N=N+K 2080 RETURN 3000 L=0 3010 IF I=4*INT(I/4)THEN L=1 3030 IF I=100*INT(I/100)THEN L=0 3050 IF I=400*INT(I/400)THEN L=1 3070 RETURN

&PR77998 S, INFERIOR SPRF718 REPRESION SPRFRIS 3.FRF7,9R REPRESION B.FRFRIGR

By Brian Woods, HV99er's

INTRODUCTION

Sprites are graphics shapes that move more smoothly than the usual characters which tend to "jump" from one screen position to another. They can be created and set in motion in one statement and will continue to move without needing to be updated unless required.

Before detailing the creation of: a sprite there are certain restrictions that must be borne in mind:-

1) In all there may be up to 28 sprites set in motion at one time

but no more than 4 can appear on any one row at a time. If more than 4 occupy the same row only the 4 sprites with the lowest numbers will be displayed.

- 2) Once a sprite is set in motion it will, unless altered as detailed later, will continue with that direction and speed until the program ends or a breakpoint is encountered.
- 3) Sprites will always pass over other characters on the screen, and where 2 or more sprites coincide, the sprite with the lowest number passes over the other sprite/s.
- 4) When a breakpoint occurs or the program stops with an error etc., sprites cease to exist and will not re-appear using CONtinue.

CREATING A SPRITE

A sprite is created by using the CALL SPRITE subprogram and is configured:-

CALL SPRITE(sprite no., ASCII character number, sprite color, dot-row, dot column, row/vertical speed, column/horizontal speed)

The first parameter, sprite number, may be any integer from 1 to 28.

The second parameter, ASCII character code may be any ASCII number between 32 and 143. Since most programmers will want to define their own sprite shape, the usual proceedure is to define a character using the CALL CHAR subprogram. By using only the ASCII numbers 33-47, 59-64 and 91-95 for your sprites you have sprites and still retain all of the alphabet and numerals for any text you may require.

The third parameter is the foreground color of the sprite and may be any integer between 1 and 16. The background color of a sprite is always 1 i.e. transparent, so that the screen color always shows through.

The fourth and fifth parameter is the starting position on the screen. Instead of using row and column as in HCHAR, VCHAR etc., the

CAL PI) mac (pi

tur

mal

1-1 192 fro

per por spr scr off

bet bot 8 t

ap; scr ro; col

> par If Spr

na: ab:

the tor po: spe ric it

spi di: gi

bu or ex CAI

> th MO

> > CA

WO

n any En 4 He 4 Will

on it tiled that the is

over and ide, mber

the tc., not

sing is

ical

rite I to

CII CII ince ine ual

ter By 47, you of

any

the and 16.

is hat

ter the and the CALL SPRITE subprogram requires the PIXEL position. (Each character is made up of 64 picture elements (pixels) which are arranged in an 8X8 grid. The character is created by turning on certain pixels and thus make a shape).

Dot-row can be any number from 1-192 (24 rows X 8 pixels per row = 192). Dot-column can be any number from 1-256 (32 columns X 8 pixels = 256). By using per column positions within these ranges a will sorite be visible on the screen. A sprite can be positioned off screen by using a row number between 193 to 256 which is off bottom of the screen.

As a guide, the pixel number is 8 times the usual row and column numbers used in HCHAR and VCHAR etc., thus row 12 col 16 is approximately the centre of the screen and in pixel form is:row 12 X 8 = 96
col 16 X 8 = 128

The sixth and seventh parameters, velocity, are optional. If not specified when creating the sprite, it will remain stationary. The velocities must be within the range -128 to 127. The larger the absolute value the faster the sprite moves.

A positive row velocity moves the sprite down, a negative moves it towards the top of the screen. If a positive column velocity is specified the sprite moves left to right and a negative velocity moves it right to left. If both the row and column are not 0 (zero) the sprite will move at an angle in the direction determined by the values given.

A typical CALL SPRITE statement would appear:CALL SPRITE(#1,33,16,96,128,50,-1)
but any numerical variable, function or expression may be used, for example:CALL SPRITE(#X,ASC(A),X/2,3+Y,10+X,50,30)
would be a valid command.

Next issue we go on to altering the sprite by using the subprograms MOTION, LOCATE, COLOR, MAGNIFY and detecting coincidence by the use of CALL COINC.

SAGREFEGE

Ŷ.

M

VIRGIN

This article by D.G. Blair originally appeared in the September 1985 edition of the Adelaide Texas Instruments Computer Club (ATICC).

Hi! My name is Don.

I recently purchased a TI-99/4A second hand. A few cartridges and a couple of old ATICC magazines came along with it. I decided to contact the club as I was finding it difficult, not knowing anyone else with a 99, to obtain software. I also thought I could pick up some tips!

I rang the co-ordinator and he advised me that the next meeting was the A.G.M., which I was welcome to attend. I felt a little uneasy about attending as 6 months earlier a computer was a new and mystifying object to me. All I knew at that time was that a computer powerful tool filled with chips and made from the same substance they use to restructure breasts. Still. I'd decided to attend.

The day of the meeting I found myself in a primary school classroom full of strangers a couple of computers with programs up and running. After committee had arrived the meeting was bought to order. The committee tried unsuccessfully to vacate their positions. When nominations were called for, there was a deathly silence. There I was at my first club meeting, not having paid any subscription, not even a member yet and I could have nominate for any one of the positions available.

I sat there astonished, until I realised the amount of that these men unselfishly parted with for benefit and survival of the club. something in. I read about a few The editor made it very clear that commands like PRINT and GOTO. he was fed up, he'd had enough. He I tried typing something like: was virtually writing the club magazine on his own. There was a PRINT NUMBERS 1 TO 10 distinct lack of INPUT from members, most of whom had not even bothered Beeeeep: * INCORRECT STATEMENT was I couldn't the reply. to fill in a survey. help but feel sorry for him and volunteered to submit an article O.K. then lets try typing....RUN occassionally.

The important message had been clearly made. If the club is to O.K. then try typing.... WHY NOT survive it needs your support and mine! So get off your buns and Beeeeep: * INCORRECT STATEMENT attend the meetings, phone, write or whatever!

To me, the main reason for having the club is to exchange information! with your mouth shut and your pen in books magazine is articles pocket! The obviously the best medium to use, as "Artificial Intelligence". into te demonstrations. your pen! Write, Ask, Tell! What you that the computer hadn't it!

One member at the meeting had different the courage to bare his soul and desperately to hold on to said he really couldn't find any glimmer of hope! rational reason why he kept playing around with his computer anyway. Have you ever faced that question? I pondering the question of why I was was almost in a state of shock, as hooked on this machine and what evil suddenly I realized that I had no really lurked beneath those QWERTY idea why it was that I habitually keys. Why was it? sit tapping away at the keyboard till the early hours of the morning.

I first brought this alien object my innocence slipping I felt like a 4 year old Christmas unpacked the box.

looked innocent sitting wired it to the tele. Then slotted in a module called PARSEC. The signs were already there. I've always been a games addict, be it cards, cricket, golf, anything at all.

The signs were even time and evident when I opened the beginners had book on BASIC. I merely browsed the through, as I couldn't wait to

Beeeeep: * CAN'T DO THAT , the reply.

This wasn't going to be as simple as I'd first thought!

I found myself going off to the This cannot be done library and taking out half a dozen at a time. One of the Ī read referred ta That I at the meetings everyone is tuned thought was reassuring! My heart Out with overflowed with glee! Did this mean really out of the club can only be beaten me 20 games to 1 on the proportional to what you put into beginners level of VIDEO CHESS? That somehow it was all an illusion? My mind flew off into a thousand tangents as I tried

The next day as I was still

I'd come to realize that the more time I spent with this My mind drifted back to the day seductive creation, the more I felt away. T through the door. I pretended all searched all the deeper recesses of the while that it was money well my mind, PEEKing at this and POKEing spent on my 4 year old's education. at that (My memory obviously wasn't on up to a simple CALL statement!). morning, as I eagerly suddenly struck upon the answer! I was amazed! Could it be? Was I really addicted to self abuse? Could enough I cure myself of this nasty habit? there. I plugged it in and If I could cure it, did I really I want to?

Change hands at "99"

1.00 34

4.0 - A.

DON

more ners wsed type few

Then

was

bly.

as

zen the to it I art

the

ean 1117 the hat My

and ied ome

i 1 1 Has vi1

RTY

the his lt Ι

of ng t

I

I Ι l d

t? 19

"JINGLE BELLS"

THIS PROGRAMME CAME TO US FROM HOLLAND VIA TONY MCGOVERN, AND ANYBODY WITH A SPEECH SYNTHESIZER AND TERMINAL EMULATOR TWO IT SINGS "JINGLE BELLS", WITH A HALTING DUTCH DON'T BE FRIGHTENED BY ALL THE DATA STATEMENTS, ACCENT. TAKING IT SLOWLY I MANAGED TYPE TO IT IN WITH ONLY MISTAKE (A MIRACLE FOR ME).

IT TAKE THE COMPUTER QUITE A LONG TIME TO READ ALL THOSE DATA STATEMENTS ALSO, SO IF IT SEEMS AS THOUGH THE COMPUTER LOCKED UP IT PROBABLY HASN'T!!!!.



100 REM *********** 110 REM * JINGLE BELLS 120 REM * for 130 REM *Terminal Emulator* 140 REM * and 150 REM *Speechsynthesizer* 160 REM * Robert Kat 170 REM * (1985)180 REM ************ 190 DATA 252,34,93,38,81,70, 126,86,34,74,120,127,93,38,8 1,70,126,86,34,74,120 200 DATA 127,93,38,81,252,29 ,70,252,39,126,31,73,252,36, 126, 124, 20, 252, 34, 83, 58, 127, 210 DATA 252,32,126,127,43,1 17,84,27,112,115,69,78,127,3 8,112,38,120,252,34 220 DATA 112,68,82,49,88,127 ,48,78,252,35,36,252,36,84,2 7,78,117,42,82,100 230 DATA 24,252,34,109,20,78 ,252,36,120,73,58,127,252,29 ,117,58,127,127 240 DATA 252,34,93,38,81,70, 126,86,34,74,120,127,93,38,8 1,70,126,86,34,74,120 250 DATA 127,93,38,81,252,29 ,70,252,39,126,31,73,252,36, 126,124,20,252,34,83,58,127, 127 260 DATA 252,32,127,43,117,8 4,27,112,115,69,78,127,38,11 2,38,120 270 DATA 252,34,112,46,82,49 ,66,127,4,78,20,127,252,29,8 4,27,78,126

280 DATA 117,31,82,120,126,2 52,32,43,252,36,109,20,20,78 ,126,252,39,120,73,58,127,12 300 DATA 252,46,82,49,252,34 ,88,38,81,252,36,124,82,68,2 52,39,124,20 301 DATA 252,46,120,78,67,12 7,127 310 DATA 48,78,20,84,3,78,25 2,34,118,42,120,126,252,36,4 3,252,39,109,20,78,126 320 DATA 252,43,120,73,58,12 7,127,127 330 DATA 43,82,126,252,32,95 ,20,126,252,34,115,54,112,12 0,126,252,36,83,25,126,252,4 0,91,67 340 DATA 127,127,127,252,29, 73, 26, 26, 26, 116, 38, 81, 252, 32 ,31,73,126,124,20,126 350 DATA 252,34,83,58,127,12 7,127 360 DATA 252,46,86,34,75,120 ,252,34,31,78,252,36,86,31,8 6,252,39,112,58,73,120,252,4 6,82,38,81,127,127 370 DATA 76,58,252,34,106,38 ,81,126,252,36,120,109,38,25 2,39,82,38,112,120 380 DATA 252,43,86,82,49,112 ,127,127,43,126,252,32,84,3, 112 390 DATA 126,252,34,115,69,7 8,126,252,36,38,112 400 DATA 126,252,29,38,120,1 26,112,14,126,120,38,81,126, 20,126,252,26,120,73,58,126





410 DATA 252,29,85,38,81,252 ,32,120,31,81,126,252,36,112 ,44,126,252,39,78,49,112,127 ,127 420 DATA 252,29,85,34,34,34, 34,34,34,34,34,34,34,34 430 DATA 252,34,93,38,81,70, 126,86,34,74,120,127,93,38,8 1,70,126,86,34,74,120 440 DATA 127,93,38,81,252,29 ,70,252,39,126,31,73,252,36, 124,124,20,252,34,83,58,127, 127 450 DATA 252,32,126,127,43,1 17,84,27,112,115,69,78,127,3 8,112,38,120,252,34 460 DATA 112,68,82,49,88,127 ,48,78,252,35,36,252,36,84,2 7,78,117,42,82,100 470 DATA 24,252,34,109,20,78 , 252, 36, 120, 73, 58, 127, 252, 29 ,117,58,127,127 480 DATA 252,34,93,38,81,70, 126,86,34,74,120,127,93,38,8 1,70,126,86,34,74,120 490 DATA 127,93,38,81,252,29 ,70,252,39,126,31,73,252,36, 126, 124, 20, 252, 34, 83, 58, 127, 127 500 DATA 252,32,127,43,117,8 4,27,112,115,69,78,127,38,11 2,38,120 510 DATA 252,34,112,46,82,49 ,88,127,4,78,1,127,252,29,84 ,27,27,27,27,78,126 520 DATA 117,31,31,31,82,120 ,127,252,32,43 530 DATA 252,36,109,20,20,20 ,20,78,126,127,252,39,120,73 ,58 540 DATA 127,127,118,58,127, 127,118,58,127,58,126,58 550 DIM A#(255) 560 CALL CLEAR 570 PRINT " PATIENCE PLE ASE .. ": : 571 PRINT Scraping my thr oat....": : : : : : : 580 FOR I=1 TO 219



590 READ A 600 A\$(I) = CHR\$(A) 610 C\$(1)=C\$(1)&A\$(I) 620 NEXT I 630 RESTORE 301 640 FOR I=1 TO 219 650 READ A 660 A\$(I)=CHR\$(A) 670 C\$(2)=C\$(2)&A\$(I) 680 NEXT I **690 RESTORE 430** 700 FOR I=1 TO 219 710 READ A 720 A\$(I)=CHR\$(A) 730 C\$(3)=C\$(3)&A\$(I) 740 NEXT I



741 REM ************ 742 REM * TO DISK 743 REM ************** 750 REM OPEN #2:"DSK1.JINGLE ,SEQUENTIAL,OUTPUT,INTERNAL ,FIXED 220 760 REM FOR I=1 TO 3 770 REM PRINT #2:C\$(I) 780 REM NEXT I 790 REM CLOSE #2 800 CALL CLEAR 810 PRINT JINGLE BEL 820 OPEN #1: "ALPHON", INTERNA 830 PRINT #1:C\$(1) 840 PRINT #1:C\$(2) 850 PRINT #1:C\$(3) 860 CLOSE #1



860 CLOSE #1
870 CALL CLEAR
880 PRINT "One more time? Pr
ess ENTER"
890 CALL KEY(O,K,S)
900 IF S=0 THEN 890
910 GOTO 820



MICROI public grade: power graph learn teach quick and conce;

Each curridevelded advandespect actives

Lower exploi upper move Worksi approj

LOGO and MATH



MICROQUESTS is a quarterly publication for classroom teachers. grades 4-8. who want to tap the of LOGO Dower beyond simple graphics. Rather than simply LOGO for LOGO's sake, learning teachers and students can quickly into exploring facinating and sophisticated mathematical concepts from multiple perspectives.

Each project includes linkages to curriculum requirements and i s developed from beginning to advanced level. delving into concepts with many activities and ways to explore those concepts.

Lower grades and slower students may explore beginning activities while upper grades and advanced students move to more advanced levels. Worksheets are provided where appropriate.

While LOGO is not the end-all answer to education, it is a powerful tool that has hardly even been explored. It is particularly efective in cultivating mathematical thinking.

Logo has been touted as a powerful tool, one which can help develop critical thinking skills. It's a language which is said to have "no floor, no ceiling." Where does that leave the average classroom teacher who's been given no tools to work with beyond those for teaching simple turtle graphics? If you're like most classroom teachers and curiculum coordinators, it probably leaves you frustrated.

As Seymour Papert, the developer of LOGO, said at LOGO '04, "Where are all the powerful ideas?"

In the January issue of Classroom Computer Learning, Papert said,

...you shouldn't be surprised that we find we know much less about what we do in the second and third year than in the first year. I think we know how to make it work. But I'm sure we're going to have to learn an awful lot about how to use it, how to develop it, how to integrate it into the learning environment at more advanced levels.

One of the beauties of LOGO (and another potentially frustrating aspect to teachers) i S its A wide variety of flexibility. students may use LOGO successfully, and as they do, they move to many different levels of proficiency and understanding. How do you offer something challenging to advanced and at the same time, students something meaningful and encouraging to slower students? And how do you integrate LOGO into demanding curriculum.

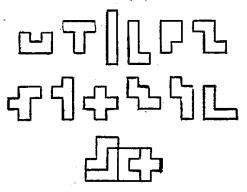
While LOGO cannot be tied to a curriculum in the sense of certain things being required on certain days or in certain grades, through the careful selection of significant projects, we can encourage and allow students to work at various levels and we can address various curriculum requirements.

MICROQUESTS will be published quarterly: JUNE, SEPTEMBER, DECEMBER and MARCH (to match American school year). Each issue will comprise 50 pages.

In Issue No. 1, you are shown how to use Transformers, Pentominoes, Bouncing Balls and Recursive Tiles to explore RATIO and PROPORTION from many different angles.

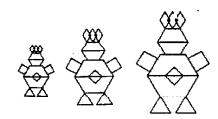
PENTOMINOES - "A LOGO costruction set"

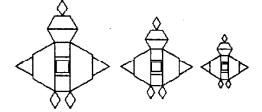
Pentominoes are shapes which cover five connected squares. There are twelve such pentominoes. Combine in as many different ways as possible, always striving for the cleanest, simplest connecting procedure.



TRANSFORMERS - "More than meets the eye"

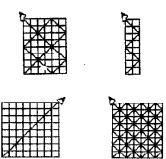
Use squares, triangles, parallelograms and trapezoids to form various robots and vehicles. In addition to the challenge of connecting shapes, experience the proportional relationships within each figure.





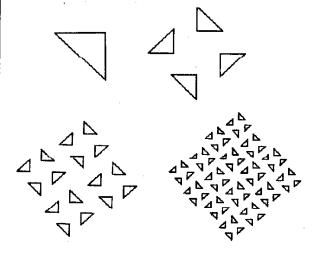
BOUNCING BALLS - "Six ball in the corner pocket"

Bounce the turtle on the grids of different proportions to discover paterns. Learn to predict which corner the turtle will exit the grids.



RECURSIVE TILES - "Of tiles and glue and disessa, too"

Experince proportions as you explore recursive patterns. The idea for this adventure came from TURTLE GEOMETRY by Abelson and diSesa. Its complex but worth the struggle.



ABOUT THE AUTHORS: Kathleen Martin, Ph.D, and Donna Bearden have an international reputation as LOGO educators. With six LOGO related books published by Reston Publishing Co., three project booklets published Martin-Bearden, Inc., articles in several publications, and teacher training and workshops throughout U.S.A., they are well qualified to take on the task of developing curriculum-linked LOGO materials. Dr. Martin has 20 years of teaching experience from elementary through the college level. Bearden has 14 years of experience editing publications developing training materials.

For details on how to go about subscribing to MICROQUESTS, Vol 2 and back issues of Vol. 1 write to:

Martin-Bearden, Inc. 2502 Peach Tree Ln. McKinney TX75069 THIS IT'S MAGA SOUR DESE IT.

Sinc disp 20 grou scre back cole fors the

tape C91



THIS PROGRAME WAS FIRST PUBLISHED IN BY COMPUTE ORIGINAL FORM ONE OF THE BEST MAGAZINE, STILL TI USER'S AND FOR THE DESERVES ALL THE SUPPORT WE CAN GIVE IT.

Since the TI-99/4A screen can display only 10 frames at once, the frames are divided into two groups (1-10 and 11-20) on alternate Press the N key to switch screens. back and forth. You can change the color of the screen background or foreground drawing color by pressing the B or F keys. Animator 2 can ! save animation files on disk or When saving on tape, enter ********************************** tape. loading from disk, be sure to frame in the sequence.

type prefix DSK1 the the beginning of each filename.

DRAWING AN IMAGE *********************

When you run animator, it displays an editing screen with 10 numbered frames. You can draw as many as 20 pictures, one in each frame, then through the frames to flip rapidly create animation. The frame number displayed at the upper left of the you're screen shows which frame currently working on. Normally, Animator begins the animation with frame 1 and ends with frame 20. you can start and end the animation wherever you like. The frame number is controlled by pressing the right and left arrow keys. The frame number also determines which frame you'll be working on when you go to the editing screen.

Let's start with a simple example. Make sure the frame number is set to 1, then press the 2 key to select the editing function and press Enter at the next prompt. After a brief pause, Animator displays a drawing grid with a blinking cursor. Edit mode has three main functions, selected by pressing different keys. Press D to draw with the cursor, E to erase and M to move the cursor without disturbing anything on the screen.

Draw a simple shape on the grid become familiar with these basic functions. As you'll see, Animator displays the shape in its actual size to the left of the drawing grid. It will also display the char definition for you to jot down. inverse function lets you reverse everything on the grid - every dot becomes a blank, and vice versa (be patient - it takes Animator about a minute to complete this process).

Once the picture is finished, you can press 7 to save it and return to the main screen. Note that you must save a picture with 7 to put it in the frame. If you exit the edit mode by pressing Q, the new picture is lost and Animator uses whatever that frame previously contained. When you return to the main screen, Animator displays the picture in

FRAME BY FRAME

CS1 for the filename. When saving Now you're ready to draw the next

lore for RTLE Its

isa,

onna

7 40

onal With kd by ject Ьу in cher hout

d to ping als. hina hool

Ms. ence and

bout 2 to:

you'll want to make only slight changes from one frame to the Τo next to simulate smooth motion. save time, Animator lets you copy a picture from one frame to another. Let's demontrate this by copying the picture from frame 1 to frame 2. Set the picture number to 1 with the arrow keys, then press 2 to edit. Animator displays a prompt, inviting you to enter a frame number. To edit the current picture number, you would just press Enter. However, by entering a different number you can copy the current picture into a different frame, then change that picture to make the next frame in your cartoon. When you enter 2 at prompt, Animator copies the the picture from frame 1 into drawing grid. When the drawing grid the appears, make some change in picture to distinguish it from frame Now press 7 to save the picture frame 2 and return to the main bath Animator displays screen. pictures in their respective frames.

After drawing a few frames, you're ready to bring them to life. The to specify the first step i 5 starting and ending frame numbers. The starting number determines which frame begins the animation, and the ending number tells Animator where the series ends. Set the starting number first. Use the arrow keys to set the frame number to 1, then press the 3 key. Now use the arrow keys to make the frame number match last frame that contains a picture, then press the 4 key. This sets the ending number. You must always set the starting and ending numbers before selecting animation don't, Animator flip⊆ (if you through all 20 frames whether they contain pictures or not). Once these numbers are set, press the 1 key to view the sequence. Press the space bar to pause and enter to stop it.

By selecting different speed and values, you can move the pause animated figure across the screen. The speed value can range from -15 to 15. When it is 0, the figure is animated in place; positive values move the figure from left to right, and negative values move it from The greater the right to left. value, the faster the figure moves. Press there 5 key to decrease the animation speed, and 6 to increase it.

The pause value controls the time delay between each frame of the animation. A small pause value makes the pictures change very quickly, while larger values slow down the process.

MACRO EDITING FEATURE

Animator provides a few macro (large scale) aditing features to help you work with longer cartoons. The insert function lets you insert a blank frame anywhere in the series. To use it, set the frame where you want to insert a blank, then press the 1 key. The designated picture (and all those following it) are bumped forward one frame. Note that the picture in frame 20 is always lost when you insert.

The delete function lets you delete any frame in the series. Change the picture number to the frame you want to eliminate, then press D. All the higher numbered pictures move down one frame, deleting the picture in the designated frame. Frame 20 is always blank after a deletion. The inverse function (press 9) works just like inverse in editing mode, but inverts all 20 frames at once.To clear all 20 frames, press Q to quit or C to clear. Since these last two functions can have drastic results. Animator lets you abort either without harm.

When you finish a sequence, press S to save it on disk. The screen clears and displays three options: You can Press A to abort the save, F to list the picture files on that disk, or any other key to continue Picture filenames with the save. are limited to eight characters (the first character cannot be a number). three-character add а Do not Animator automatically extension; appends the extension ANI when you save or load a picture file.

Once you have the Animation right plus movement and speed, if you have a printer and want a hard copy just follow the instructions on screen if not its not to hard to change it around so that it will print to the screen. In our next we will put in a sme, that will accept smali issue your programe, saved files. This is to help build a background or scene to put some real life into your animation.

PETER COXON HV99.

time the value very slow

large
you
The
ries.
you
press

) are

that

lways

elete
e wanhe
town
re is
worde,
ce.ui
ults,
ults,e

essee: verns: Fteesee. tinnesee tina(t) ertlly u

right
have
just
the
will
next
small
your
build

ANEMATOR 2.

10 FORE=2 :: BACK=12 :: 20 CALL CLEAR :: GOSUB 480 : : F=1 :: L=10 :: N=1 :: BE=1 :: E=20 :: SP=0 :: DL=0 :: FOR I=104 TO 143 :: CALL CHA R(I,RPT\$("0",16)):: NEXT I 30 CALL MAGNIFY(4):: DIM B(1 6,16),C\$(15),E\$(20),IFLAG(20):: Q=="DEVICE(DSK1.FILE OR CS1)?" :: GOSUB 910 :: CALL SCREEN(BACK):: GOSUB 970 40 CALL KEY(0,K,S):: IF S=0 THEN 40 50 IF K>48 AND K<57 THEN K=K -48 :: ON K GOTO 170,500,220 ,220,240,240,250,250 35 IF K-80 THEN GOSUB 2000 60 IF K=66 THEN GOSUB 260 70 IF K=70 THEN GOSUB 270 80 IF K=73 THEN GOSUB 280 90 IF K=69 THEN GOSUB 300 100 IF K=65 THEN GOSUB 330 110 IF K=76 THEN GOSUB 370 120 IF K-67 THEN GOSUB 390 130 IF K=81 THEN END 140 IF K=78 THEN GOSUB 410 150 IF K=68 OR K=83 THEN N=N -({N<>1}-(N=1)*19)*(K=83)+((N(>20) - (N=20) + 19) + (K=68) :: DISPLAY AT(4,1):N; 140 GOTO 40 170 CALL CLEAR :: GOSUB 480 :: FOR I=5 TO 8 :: CALL COLO R(I, FORE, 1):: NEXT I :: FOR I=1 TO 20 :: CALL CHAR(60+1* 4,E\$(I)):: NEXT I 180 GOSUB 490 :: CALL SPRITE (#1,60+BE#4,FORE,100,100);; CALL MOTION(#1,0,SP%6) 190 FOR I=BE TO E :: CALL PA TTERN(#1,60+1*4):: DISPLAY A T(1,1):I :: FOR J=1 TO DL :: NEXT J :: NEXT I 200 CALL KEY(0,K,S):: IF S=0 THEN 190 ELSE CALL DELSPRIT E(#1):: CALL CHARSET 210 FORE=FORE-1 :: GOSUB 270 :: FOR I=5 TO 8 :: CALL COL OR(I,2,1):: NEXT I :: LF=1 ; : GOSUB 910 :: GOSUB 970 :: **GOTO 40** 220 IF K=3 THEN BE-N ELSE E-230 GOSUB 1060 :: GOTO 40 240 SP=SP-(SP<>-15) *(K=5)+(S P(>15)*(K=6):: GOSUB 1060 :: GOTO 40

250 DL=DL-(DL<>0)*(K=7)+(DL= 0) * (K=7) * 150 + (DL<)150) * (K=8)-(DL=150)*(K=8)*150 :: GQSUB 1060 :: GDTO 40 260 BACK=BACK+1+(BACK=16) *14 :: CALL SCREEN(BACK):: RETU RN 270 FORE=FORE+1+(FORE=16) *15 :: FOR I=10 TO 14 :: CALL C OLOR(I, FORE, I):: NEXT I :: C ALL COLOR(9, FORE, 16):: RETUR 280 GOSUB 320 :: IF A\$="N" T HEN RETURN 290 FOR I=20 TO N+1 STEP -1 :: E#(I)=E#(I-1):: NEXT I :: E\$(N)=RPT\$("0",64):: GOSUB 450 :: GOSUB 440 :: RETURN 300 GOSUB 320 :: IF A="N" T HEN RETURN 310 FOR I=N TO 19 :: E\$(I)=E \$(I+1):: NEXT I :: E\$(20)=RP T\$("0".64):: GOSUB 450 :: GO SUB 440 :: RETURN 320 DISPLAY AT(12,4): "ARE YO U SURE(Y/N)?" :: ACCEPT AT(1 2,24) VALIDATE("YN"): A\$:: IF A#="Y" THEN RETURN ELSE GOS **UB 440 :: RETURN** 330 GOSUB 420 :: OPEN #1:D\$, INTERNAL, OUTPUT, FIXED 192 :: PRINT #1:FORE, BACK :: FOR I =1 TO 20 STEP 2 331 IF I=19 THEN 333 ND(E#(I+1)=RPT#("0",64))THEN 334 333 PRINT #1:E\$(I),E\$(I+1),I 334 NEXT I 335 PRINT #1: IFLAG(1), IFLAG(2), IFLAG(3), IFLAG(4), IFLAG(5), IFLAG(6), IFLAG(7), IFLAG(8) , IFLAG(9), IFLAG(10) 336 PRINT #1: IFLAG(11), IFLAG (12), IFLAG(13), IFLAG(14), IFL AG(15), IFLAG(16), IFLAG(17), I FLAG(18), IFLAG(19), IFLAG(20) 350 CLOSE #1 :: FOR I=10 TO 14 :: CALL COLOR(I, FORE, 1):: NEXT I :: CALL SCREEN (BACK) 340 IF ASC(D\$)=67 THEN GOSUB 970 :: RETURN ELSE GOSUB 44 O :: RETURN 370 GOSUB 420 :: OPEN #1:D\$, INTERNAL, INPUT , FIXED 192 :: INPUT #1:FORE, BACK 371 FOR I=1 TO 20 STEP 2 372 INPUT #1:E\$(I),E\$(I+1),I

373 NEXT I 374 INPUT #1: IFLAG(1), IFLAG(2), IFLAG(3), IFLAG(4), IFLAG(5), IFLAG(6), IFLAG(7), IFLAG(8) , IFLAG(9), IFLAG(10) 375 INPUT #1: IFLAG(11), IFLAG (12), IFLAG(13), IFLAG(14), IFL AG(15), IFLAG(16), IFLAG(17), I FLAG(18), IFLAG(19), IFLAG(20) 380 GOSUB 450 :: GOTO 350 390 GOSUB 320 :: IF A#="N" T HEN RETURN 400 FOR I=1 TO 20 :: E\$(I)=R PT\$("0",64):: NEXT I :: GOSU B 450 :: GOSUB 440 :: RETURN 410 F=F-(F=1)*10+(F=11)*10 : L=F+9 :: GOSUB 450 :: GOSU B 460 :: RETURN 420 DISPLAY AT(12,1):Q# :: A CCEPT AT(13,1):D\$:: IF D\$=" THEN GOSUB 440 430 RETURN 440 CALL HCHAR(12,1,32,64):: RETURN 450 J=0 :: FOR I=F TO L :: C ALL CHAR(104+J*4,E\$(I)):: J= J+1 :: NEXT I :: RETURN 460 CALL HCHAR(9,2,32,2):: J =F+1 :: FOR I=2 TO 26 STEP 3 :: DISPLAY AT(9,I):J :: J=J +1 :: NEXT I :: F\$=STR\$(F):: DISPLAY AT(4,1):N; 470 FOR I=1 TO LEN(F\$):: CAL L HCHAR(9,1+I,ASC(SEG#(F#,I, 1))):: NEXT I :: RETURN 480 DISPLAY AT(1,7): "PLEASE WAIT..." :: RETURN 490 CALL HCHAR(1,9,32,14):: RETURN 500 DISPLAY AT(12,1): "TYPE 2 1 TO ABORT, A # 1-20 OR (ENTE R> FOR CURRENT #:" 510 ACCEPT AT(13,27):A# :: I F AS="" THEN PN=N :: GOTO 56 520 IF A\$<"1" OR A\$>"9" THEN 510 530 J=VAL(A\$):: IF J=21 THEN GOSUB 440 :: GOTO 40 540 IF J>20 OR J<1 THEN 510 550 IF J=N THEN PN=N ELSE PN =.7 560 IFLAG(PN)=IFLAG(N):: CAL L CLEAR :: DISPLAY AT(2,1):" 1-MOVE 2-DRAW 3~ERASE" : : DISPLAY AT(3,1): "4-CLEAR 5 -INVERSE 6-DISPLAY"

570 DISPLAY AT(4,1):"7-SAVE FRAME=";PN :: DISP 8-QUIT LAY AT(5,2): "(USE ARROW KEYS TO MOVE)" :: GF=1 :: GOSUB 920 580 IF IFLAG(N)=1 THEN CALL CHAR(100, U\$, 101, "") ELSE CALL CHAR (100, "", 101, U\$) 590 F\$=E\$(N):: FOR I=0 TO 63 STEP 2 :: D=ASC(SEG\$(F\$,I+1 .1))-48 :: D=D+(D>9)*7 :: D1 =ASC(SEG\$(F\$,I+2,1))-48 :: D 1=D1+(D1>9)*7 600 D=-15*(IFLAG(PN)=1)+D*(I FLAG(PN)=1)-(IFLAG(PN)=0)*D:: D1=-15*(IFLAG(PN)=1)+D1*(IFLAG(PN)=1)-(IFLAG(PN)=0)*D 610 DISPLAY AT(6+1/2+(1)31)* 16,3-(1>31)*8):C\$(D):: DISPL AY AT(6+1/2+(1>31)*16,7-(1>3 1) *8) : C\$(D1) :: NEXT I 620 CALL SPRITE(#28,96,14,41 ,33):: KHAR=101 :: FLAG=1 :: R=1 :: C=1 630 CALL KEY(0,K,S):: IF S=0 THEN 630 ELSE CALL DELSPRIT E(#1) 640 IF K=51 THEN KHAR=100 :: FLAG=0 :: GOTO 790 450 IF K=49 THEN FLAG=1 :: G **070 800** 660 IF K=50 THEN KHAR=101 :: FLAG=0 :: GOT0 790 670 IF K<>52 THEN 690 680 FOR R=1 TO 16 :: CALL HC HAR(S+R,5,100,16):: NEXT R : : DFLAG=0 :: GOTO 620 690 IF K<>53 THEN 720 700 DFLAG=0 :: IFLAG(PN)=-(I FLAG(PN)=0):: IF IFLAG(PN)=0 THEN CALL CHAR(100, "", 101, U \$)ELSE CALL CHAR(100,U\$,101, 710 GOTO 800 720 IF K=56 THEN CALL DELSPR ITE(ALL):: LF=1 :: GOSUB 910 :: GOSUB 970 :: GOTO 40 730 IF K=66 THEN GOSUB 260 : : DFLAG=1 740 IF K=70 THEN GOSUB 270 : : DFLAG=1 750 IF K=54 OR K=55 THEN CAL L DELSPRITE(ALL):: IF DFLAG= 1 THEN 890 ELSE 810 760 C=C+(K=83)+((C=1)*(K=83) *16) - (K=68) - ((C=16) * (K=68) *1

1

(A

770 R=R+(K=69)+((R=1)*(K=69) *16)-(K=88)-((R=16)*(K=88)*1 6) 780 CALL LOCATE(#28,8*R+33,8 *C+25):: IF FLAG=1 THEN 800 790 DFLAG=0 :: CALL HCHAR(5+ R,4+C,KHAR) 800 CALL SOUND(20,200,5):: G OTO 630 810 GOSUB 480 :: FOR R=1 TO 16 :: FOR C=1 TO 16 :: CALL GCHAR (5+R, 4+C, GC):: GC=GC-10 O :: B(R,C)=GC :: NEXT C :: NEXT R 820 F*="" :: IF IFLAG(PN)=0 THEN HEX#="0123456789ABCDEF" ELSE HEX#="FEDCBA9876543210 830 FOR R=1 TO 16 :: LOW=B(R ,5)*8+B(R,6)*4+B(R,7)*2+B(R, 8)+1 840 HIGH=B(R,1) *8+B(R,2) *4+B (R,3)*2+B(R,4)+1850 F==F=&SEG=(HEX=,HIGH,1)& SEG\$(HEX\$,LOW,1):: NEXT R :: FOR R=1 TO 16 860 LOW=B(R,13)*8+B(R,14)*4+ B(R, 13) #2+B(R, 16) +1 870 HIGH=B(R,9)*8+B(R,10)*4+ B(R,11) *2+B(R,12)+1 880 F==F=&SEG=(HEX=,HIGH,1)& SEG\$(HEX\$,LOW,1):: NEXT R :: DISPLAY AT(22,1):F# :: IF K =54 THEN DFLAG=1 890 IF K=35 THEN E=(PN)=F= : : LF=1 :: GOSUB 910 :: GOSUB 970 :: GOTO 40 900 CALL MAGNIFY(4):: CALL C HAR (36,F\$):: GOSUB 490 :: CA LL SPRITE(#1,36,FGRE,80,175) :: GOTO 620 910 CALL CHAR (33, RPT\$ ("81",8),36,RPT\$("0",14)&"FF",37,"F F"&RPT\$("0",14),38,RPT\$("01" ,8),39,RPT\$("80",8)):: IF LF =1 THEN LF=0 :: RETURN 920 U\$=RPT\$("F",16):: CALL C HAR(100, " ", 101, U\$, 96, "F09090 F"&RPT#("0",57)):: CALL COLO R(9, FORE, 16):: IF GF=1 THEN GF=0 :: RETURN 930 F\$="00000001001000110100 0101011001111000100110101011 1100110111101111 940 FOR I=0 TO 15 :: Z\$=SEG\$ (F#, IX4+1,4):: D#∞""

950 FOR J=1 TO 4 :: T=VAL(SE G\$(Z\$,J,1))+100 :: D\$=D\$&CHR \$(T):: NEXT J :: C\$(I)=D\$:: NEXT I 960 FOR I=1 TO 20 :: E\$(I)=R PT\$("0",64):: NEXT I :: RETU 970 CALL CLEAR :: DISPLAY AT (1,10): "ANIMATOR 2." :: CALL HCHAR(4,2,35):: DISPLAY AT(4,5): "B= E≡ SP= D=" 980 GOSUB 1060 :: CALL VCHAR (6,1,38,2):: CALL VCHAR(6,31 ,39,2):: FOR I≈2 TO 29 STEP 3 :: CALL HCHAR(5, I, 36, 2):: CALL HCHAR(8,1,37,2) 990 NEXT I :: FOR I=4 TO 28 STEP 3 :: CALL VCHAR(6,I,33, 2):: NEXT I :: GOSUB 460 :: GOSUB 450 :: J=104 1000 FOR I=2 TO 29 STEP 3 :: CALL HCHAR(6,I,J):: CALL HC HAR(6, I+1, J+2):: CALL HCHAR(7, I, J+1):: CALL HCHAR(7, I+1, J+3):: J=J+4 :: NEXT I 1010 CALL HCHAR(15,1,49):: C ALL HCHAR(16,1,52):: CALL HC HAR (17, 1, 55) 1020 DISPLAY AT(15,1): "ANIMA TE 2 EDIT 3 START PIC" :: DISPLAY AT(16,1): "END PIC 5 -SPEED 6 +SPEED" 1030 DISPLAY AT(17,1):"-SWIT CH 8 +SWITCH 1040 DISPLAY AT(20,1):"(I)NS ERT PIC D(E)LETE PIC" :: DI SPLAY AT(21,1): "S(A)VE (C)LEAR ALL' AD 1050 DISPLAY AT(22,1):"(N)EX (Q)UIT":TAB(8); T GROUP *(P)RINT OUT* :: RETURN 1060 DISPLAY AT(4,7):BE;:: D ISPLAY AT(4,13):E;:: DISPLAY AT(4,20):SP;:: DISPLAY AT(4 ,26):STR\$(DL)&SEG\$(" ",1,3 -LEN(STR\$(DL)));:: RETURN

2000 ! PRINT OUT
2010 OPEN #2:"PIO"
2020 DISPLAY AT(10,5):"PRINT
ING DATA" :: PRINT #2:CHR\$(1
4);"ANIMATOR PROGRAMME AND D
ATA": ::
2030 PRINT #2:"VARIABLES B
E=";BE;" FORE=";FORE;" E
-";E;" SP=";SP: ::
2040 PRINT #2:CHR\$(15);"CALL
MAGNIFY(4)":" FOR I=1 TO 20
":" CALL CHAR(60+I*4,E\$(

"DECK THE HALLS"

Programming music on the TI-99/4A with the CALL SOUND command can produce delightful results.

To get started, You'll need to know how to read notes and their lengths, have a copy of the music you want to reproduce and locate the musical tone frequencies chart on page 111-7 of the "User's Reference Guide."

The CALL SOUND command line consists of Duration (length of the note), Frequency (numerical equivalent of the note) and Volume (loudness or softness of the note); 100 CALL SOUND (Dur, Freq, Vol)

If you wanted to program more than one tone at the same time, the line would look like this:

100 CALL SOUND (D,F,V,F2,V2,F3,V3) Whenever you program two or more notes together, the duration remains the same and is listed only once in the line.

The duration of a note depends on the time signature of the music. The signature tells you how many beats are in each measure and is indicated at the beginning of the music. For your first program, you may want to start with music written in 4/4 signature (four beats to the measure).

To determine the duration, first chose the length of each measure. For example, if you choose 960 (a fast tempo) as the length of each measure, a quarter note would be 240 (1/4 of 960), a half note would be 480 (1/2 of 960) and a dotted half note would be 720 (3/4 of 960).

The frequency is the actual note. Frequencies for notes are listed in the "User's Reference Guide." The frequency for middle C, for example is 262. The frequency for high C is 523

Volumes range from O (loudest) to 10 (most quiet).

If you programmed a line to play midle and high C together as half notes at a fairly loud volume, the line would look like this:

100 CALL SOUND (480,262,1,523,1)

A program for "DECK THE HALLS" follows. Remember, you can make the music play faster or slower by changing the duration of each measure and of each note.

100 CALL SOUND (360,523,1,440,1)
110 CALL SOUND (120,466,1,392,1)
120 CALL SOUND (240,440,1,349,1)

130 CALL SOUND (240, 392, 1, 330, 1) 140 CALL SOUND(240,349,1,294,1) 150 CALL SOUND (240, 392, 1, 330, 1) 160 CALL SOUND (240, 440, 1, 349, 1) 170 CALL SOUND (240,349,1,262,1) 180 CALL SOUND (120,392,1,330,1) 190 CALL SOUND (120,440,1,349,1) 200 CALL SOUND(120,466,1,392,1) 210 CALL SOUND(120,392,1,330,1) 220 CALL SOUND (360,440,1,349,1) 230 CALL SOUND(120,392,1,294,1) 240 CALL SOUND (240, 349, 1, 262, 1) 250 CALL SOUND (240,330,1,262,1) 260 CALL SOUND (480, 349, 1, 262, 1) 270 CALL SOUND (360,523,1,440,1) 280 CALL SOUND(120,466,1,392,1) 290 CALL SOUND (240, 440, 1, 349, 1) 300 CALL SOUND(240,392,1,330,1) 310 CALL SOUND (240,349,1,294,1) 320 CALL SOUND (240, 392, 1, 330, 1) 330 CALL SOUND (240,440,1,349,1) 340 CALL SOUND(240,349,1,262,1) 350 CALL SCUND(120,392,1,330,1) 360 CALL SOUND(120,440,1,349,1) 370 CALL SOUND(120,466,1,392,1) 380 CALL SOUND (120, 392, 1, 330, 1) 390 CALL SOUND (360,440,1,349,1) 400 CALL SOUND(120,392,1,294,1) 410 CALL SOUND(240,349,1,262,1) 420 CALL SOUND (240,330,1,262,1) 430 CALL SOUND (480, 349, 1, 262, 1) 440 CALL SOUND (360, 392, 1, 330, 1) 450 CALL SOUND(120,440,1,349,1) 460 CALL SOUND (240,466,1,392,1) 470 CALL SOUND(240,392,1,330,1) 480 CALL SOUND (360,440,1,349,1) 490 CALL SOUND (120, 466, 1, 392, 1) 500 CALL SOUND (240,523,1,440,1) 510 CALL SOUND (240, 392, 1) 520 CALL SOUND(120,440,1,349,1) 530 CALL SOUND(120,494,1,392,1) 540 CALL SOUND (240,523,1,392,1) 550 CALL SOUND (120, 587, 1, 392, 1) 560 CALL SOUND(120,659,1,392,1) 570 CALL SOUND(240,698,1,440,1) 580 CALL SOUND (240, 659, 1, 392, 1) 590 CALL SOUND (240,587,1,349,1) 600 CALL SOUND (480,523,1,330,1) 610 CALL SOUND (360,523,1,440,1) 620 CALL SOUND(120,466,1,392,1) 630 CALL SOUND (240,440,1,349,1) 640 CALL SOUND(240,392,1,330,1) 650 CALL SOUND(240,349,1,294,1) 660 CALL SOUND (240, 392, 1, 330, 1) 670 CALL SOUND (240,440,1,349,1) 680 CALL SOUND (240, 349, 1, 262, 1) 690 CALL SOUND(120,587,1,349,1) 700 CALL SOUND(120,587,1,349,1) 710 CALL SOUND(120,587,1,349,1) 720 CALL SOUND(120,587,1,349,1) 730 CALL SOUND (360,523,1,330,1) 740 CALL SOUND(120,466,1,392,1) 750 CALL SOUND(240,440,1,349,1) 760 CALL SOUND (240,392,1,330,1) 770 CALL SOUND (480,349,1,294,1) 780 GOTO 100

FOF

HEI

EXF

MOF

ABI

THE

AM

CHI

ADI

IN

DO IT YOURSELF MODULE PORT EXPANDER

INTEREST OF OTHER FOR THE WHO REQUIRE A CARTRIDGE MEMBERS EXPANDER WITH A CAPACITY OF FOUR THE MORE CARTRIDGES AND ADVANTAGE OF PLACING THE UNIT OUT OF THE IMMEDIATE WAY OF THE CONSOLE , I AM INCLUDING DETAILS OF A UNIT WHICH I HAVE CONSTRUCTED AND WORKS ON MY REQUIRE SOME OWN TI99/4A BUT MAY ADDITIONAL CIRCUITRY TO THAT SHOWN IN THE ACCOMPANYING DIAGRAM. I WILL AS I PROCEED FURTHER BY WAY ADVISE OF THE NEWSLETTER AS TO ANY CHANGES MADE.

MY OWN UNIT HAS SIX MODULE PORTS HOUSED IN A PLASTIC CONTAINER ALONG WITH A SIX WAY TWO POLE ROTARY SELECTOR SWITCH AND MOMENTARY ON PUSHBUTTON RESET SWITCH FOR RETURNING TO THE MASTER TITLE SCREEN PRIOR TO SELECTING REQUIRED MODULE. THESE ITEMS INCLUDING SOME RESISTORS/CAPACITOR) ARE CONNECTED VIA A 36 WAY IDC CABLE TO A DISUSED AND SLIGHTLY REARRANGED (AND NO LONGER REQUIRED) MODULE WHICH PLUGS DIRECTLY INTO THE MODULE PORT.

I BUILT THE UNIT ON A COPPER STRIP BOARD AND SOLDERED ALL ITEMS ONTO THE PRECUT COPPER TRACKS. THEN WIRE WRAPPED ALL PINS BETWEEN EACH EDGE CONNECTOR. (AN ETCHED BOARD WOULD HAVE SAVED A CONSIDERABLE AMOUNT OF TIME IN CONSTRUCTION).

TWO DIAGRAMS ARE SHOWN, ONE GIVING THE PINOUTS OF THE MODULE PORT LOOKING FROM THE FRONT OF THE CONSOLE, AND THE SECOND SHOWING THE WIRING DIAGRAM.

PARTS REQUIRED

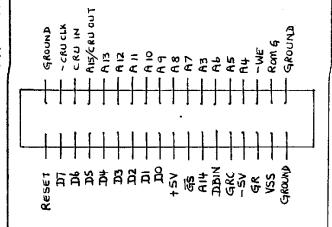
Some .01 CERAMIC CAPS 47-51K OHM RESISTORS 5% COPPER STRIP BOARD OR CIRCUIT BOARD. 36 WAY D/SIDED EDGE CONNECTERS (UTILUX H271800 GD8 OR EQUID.) 36 WAY IDC CABLE 6 WAY 2 POLE SWITCH MOM. ACTION PUSH-BUTTON CASE WITH LID WIRE FOR WIREWRAPPING ?? DISUSED MODULE "COMPLETE WITH A D/SIDE EDGE CONNECTER CARD NOTE !!! NOT ALL MODULES HAVE 36 CONTACTS ON THE **EDGE CONNECTER** SIDED UTILUX

THE DOUBLE SIDED UTILUX EDGE CONNECTORS ARE AVAILABLE FROM:
A AND B DISTRIBUTORS PTY LTD.
SUITE 107A , 1ST FLOOR
432 CHAPEL ROAD
BANKSTOWN 2200.

BANKSTOWN 2200.
TELEPHONE 02 709 7036
(COST WAS APPROX. \$5.30
EA. IN LOTS OF 10)

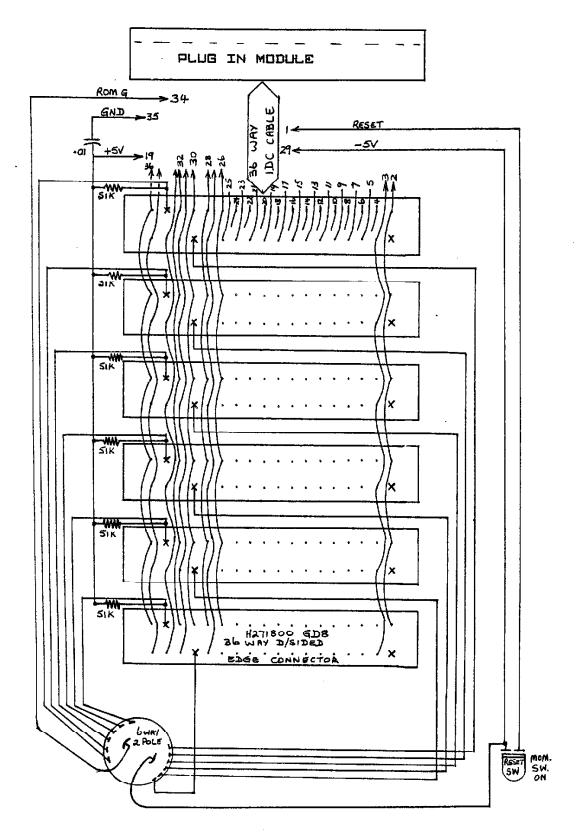
SHOULD SUFFICIENT INTEREST BE SHOWN THEN SUITABLE ARTWORK FOR A SINGLE SIDED BOARD MAY BE LOOKED INTO.

TERRY ROSS...HV99 1/11/85



LAYOUT OF PARTS FOR CARTRIDGE EXPANDER

SHOWN LOOKING AT UNDERSIDE OF BOARD



CHRISTMAS MUSIC PROGRAMME PART 2.

BY SARRY JONES NV99

OK. YOU SHOULD HAVE PART ONE KEYED IN BY NOW, IF NOT PERHAPS YOU SHOULD CONSIDER ONE OF THE OTHER PROGRAMS IN THIS ISSUE. FOR THOSE WHO HAVE COMPLETED PART ONE SIMPLY LOAD IT IN, ADD THE NEW LINE NUMBERS, TYPE RUN AND THEN SIT BACK WITH THE REST OF THE FAMILY AND ENGOY A MUSICAL AND GRAPHICS CHRISTMAS TREAT FOR ALL.

100 GOSUB 10000 170 DISPLAY AT(5,12)SIZE(6): "A VERY" :: DISPLAY AT(7,8)S IZE(15): "MERRY CHRISTMAS" :: DISPLAY AT (9,13) SIZE (4): "FR GM" 180 DISPLAY AT(12,5)SIZE(20) :"HUNTER VALLEY 99(ERS" :: D ISPLAY AT(14,10)SIZE(10): "US ER GROUP" :: DISPLAY AT(21,8)SIZE(13): "PRESS ANY KEY" 190 CALL KEY(3,KY,ST):: IF S T<1 THEN 190 200 FOR A=5 TO 26 :: CALL VC HAR(3,A,32,20):: NEXT A :: C ALL SCREEN(12) 210 DISPLAY AT(5,4)SIZE(23): "1-RUDOLPH THE RED NOSED" 215 DISPLAY AT(6,12)SIZE(8): "REINDEER" 220 DISPLAY AT(8,4)SIZE(16): "2-THE FIRST NOEL" 230 DISPLAY AT(10,4)SIZE(14) :"3-JINGLE BELLS" 240 DISPLAY AT(13,5)SIZE(21) :"A-SELECT 1 OR 2 TUNES" :: DISPLAY AT(15,5)SIZE(11): "B-ALL TUNES" :: DISPLAY AT(21, 5) SIZE(20): "PLEASE MAKE CHOI CE ?" 245 DISPLAY AT(17,5)SIZE(6): "C-EXIT" 250 ACCEPT AT(21,24)SIZE(-1) :MC\$:: IF (MC\$<"A")GR(MC\$>" C")OR(MC\$="")THEN CALL VCHAR (18,26,63):: GOTO 250 255 IF MC=="C" THEN DISPLAY AT(12,1) ERASE ALL: "A MERRY C HRISTMAS TO ALL" :: END

260 IF MC#="B" THEN TN(1)=1 :: TN(2)=2 :: TN(3)=3 :: B=3 :: GOTO 310 265 CALL SCREEN(4) 270 FOR R=11 TO 22 :: CALL H CHAR(R,5,32,24):: NEXT R :: DISPLAY AT(13,3)SIZE(24):"IF ONLY 1 TUNE IS NEEDED" 280 DISPLAY AT(15,3)SIZE(24) : "ENTER O ON SECOND PROMPT" :: FOR I=1 TO 2 :: DISPLAY A T(18,6)SIZE(14): "ENTER TUNE) ?". 290 ACCEPT AT(18,19)SIZE(-1) :N\$:: IF (N\$<"0")8R(N\$>"3") GR(N\$="")THEN CALL VCHAR(18, 21,63):: GOTO 290 300 TN(I)=VAL(N\$):: NEXT I : : B=2 310 FOR A=1 TO B :: IF TN(A) <>O THEN CALL CLEAR :: ON TN (A) GOSUB 3000,4000,5000 320 FOR DL=1 TO 500 :: NEXT DL :: Y=0 :: NEXT A 500 CALL CLEAR :: CALL SCREE N(16):: CALL CHAR(33, "010103 030FFF7F07",34,"8080C0C0F0FF FEEO",35,RPT\$("F",16)):: CAL L COLOR(1,4,16,10,7,7,9,2,2) 510 FOR A=0 TO 14 STEP 2 :: CALL HCHAR(4+A, 15-(A/2), 33): : CALL HCHAR(4+A,16-(A/2),35 ,A+2);; CALL HCHAR(4+A,18+(A /2),34) 520 CALL CHAR(36, "0000000107 OF3F7F".37."0000000C0E0F8FC "):: DISPLAY AT(3,14):"\$";"%

530 CALL HCHAR (5+A, 16-(A/2), 35, A+2):: CALL HCHAR(5+A, 15-(A/2),33):: CALL HCHAR(5+A,1 8+(A/2),34):: NEXT A 540 FOR A=19 TO 20 :: CALL H CHAR(A, 16, 104, 2):: NEXT A :: FOR A=21 TO 23 :: CALL HCHA R(A,13,96,8):: NEXT A :: CAL L HCHAR (24,14,96,6) 550 CALL MAGNIFY(2):: CALL C HAR(96, "01010303FF7F3F1F",97 ,"OF1F3F3E7870C080",98,"0000 8080FEFCF8F0*,99,*E0F0F8F83C 100602") 560 CALL SPRITE(#1,96,11,1,1 13, #2, 97, 11, 17, 113, #3, 98, 11, 1,129,44,99,11,17,129) 565 DATA 1,196,247,1,330,392 ,1,294,349,1,262,330,4,196,2 47 570 RESTORE 610 :: FOR X=5 T 0 28 :: READ C,R,COL :: CALL SPRITE(#X, 46, C, R-15, COL-8): : NEXT X 575 RESTORE 12010 :: GOSUB 1 2000 :: GOSUB 12000 580 Y=INT(3*RND)+1 :: IF Y=1 THEN A=5 :: B=4 :: C=14 :: D=16 590 IF Y=2 THEN A=4 :: B=14 :: C=16 :: D=5 ELSE A=14 :: B=16 :: C=5 :: D=4 600 FOR Z=5 TO 28 STEP 4 :: CALL COLOR(#Z,A,#Z+1,B,#Z+2, C, #Z+3, D):: NEXT Z 608 DISPLAY AT(24,3): "PRESS ANY KEY TO CONTINUE" :: CALL KEY(3,KY,ST):: IF ST(1 THEN 580 609 CALL DELSPRITE(ALL):: CA LL CLEAR :: CALL CHARSET :: GOSUB 10000 :: GOTO 200 610 DATA 12,56,136,14,122,16 0,15,120,96,5,144,128,13,144 ,184,12,80,112,14,144,88,15, 40.128 620 DATA 5,104,152,13,56,144 ,12,32,136,14,120,136,15,136 ,136,5,96,104,13,144,112,12, 72,144 630 DATA 14,56,120,15,64,128 ,5,136,168,13,92,136,12,128, 120, 14, 80, 150, 15, 104, 112 640 DATA 13,64,136 2000 DISPLAY AT (RA, CA) SIZE (2):"'"i"h" :: DISPLAY AT(RB,C B) SIZE(2): "a"; "b" :: RETURN 3000 M=325 !RUDOLPH

3010 RESTORE 3400 :: GOSUB & 000 :: X=7 :: RESTORE 3600 : : GOSUB 7000 3020 RESTORE 3410 :: GOSUB 6 000 :: X=7 :: RESTORE 3610 : : GOSUB 7000 3030 RESTORE 3420 :: GOSUB 6 000 :: X=7 :: RESTORE 3620 : : GOSUB 7000 3040 RESTORE 3430 :: GOSUB 6 000 :: X=7 :: RESTORE 3630 : : GOSUB 7000 3050 RESTORE 3440 :: GOSUB 6 000 :: X=7 :: RESTORE 3640 : : GOSUB 7000 3060 RESTORE 3450 :: GOSUB 6 000 :: X=7 :: RESTORE 3650 : : GDSUB 7000 3070 RESTORE 3460 :: GOSUB 6 000 :: X=7 :: RESTORE 3660 : : GOSUB 7000 3080 RESTORE 3470 :: GOSUB 6 000 :: X=7 :: RESTORE 3670 : : GOSUB 7000 3090 RESTORE 3480 :: GOSUB 6 000 :: X=7 :: RESTORE 3680 : GOSUB 7000 3100 RESTORE 3490 :: GOSUB 6 000 :: X=5 :: RESTORE 3690 : : GOSUB 7000 3110 RESTORE 3500 :: GOSUB & 000 :: X=7 :: RESTORE 3700 : : GOSUB 7000 :: CALL CLEAR 3120 RESTORE 3510 :: GOSUB 6 000 :: X=7 :: RESTORE 3710 : : GOSUB 7000 3130 RESTORE 3520 :: GOSUB 6 000 :: X-7 :: RESTORE 3720 : : GOSUB 7000 3140 RESTORE 3530 :: GOSUB 6 000 :: X=7 :: RESTORE 3730 : : GOSUB 7000 3150 RESTORE 3540 :: GOSUB 6 000 :: X=7 :: RESTORE 3740 : : GOSUB 7000 3140 RESTORE 3550 :: GOSUB 4 000 :: X=8 :: RESTORE 3750 : GOSUB 7000 :: RETURN 3400 DATA RU-DOLPH THE RED N OSED REIN-DEER, O 3410 DATA HAD A VER-Y SHIN-Y NOSE, 3 3420 DATA AND IF YOU EV-ER S AW IT,2 3430 DATA YOU WOULD E-VEN SA Y IT GLOWS, 2 3440 DATA ALL OF THE OTH-ER REIN-DEER, 2

3450 DATA USED TO LAUGH AND HIM NAMES, 2 3460 DATA THEY NEVER LET POO R RU-DOLPH, 3 3470 DATA JOIN IN AN-Y REIN-DEER GAMES, 2 3480 DATA THEN ONE FOGGY CHR ISTMAS EVE, 2 3490 DATA SAN-TA CAME TO SAY 3500 DATA RU-DOLPH WITH YOUR NOSE SO BRIGHT, 2 3510 DATA WON(T YOU GUIDE MY SLEIGH TO-NIGHT?,3 3520 DATA THEN HOW THE REIN-DEER LOVED HIM, 3 3530 DATA AS THEY SHOUT-ED O UT WITH GLEE, 3 3540 DATA RU-DOLPH THE RED-N REIN-DEER, 3 OSED 3550 DATA YOU(LL GO DOWN IN HIS-TO-RY.,3 3600 DATA .75,196,196,.8,220 ,220,1,196,196,1,165,165,1,2 62,262,1,220,220,3,196,196 3610 DATA .5,196,247,.5,220, 262, .5, 196, 247, .5, 220, 262, 1, 196,247,1,262,330,4,247,294 3620 DATA .65,175,220,.5,196 ,247,1,175,220,1,147,175,1,2 47,294,1,220,262,3,196,247 3630 DATA .5,196,247,.5,220, 262, .5, 196, 247, .5, 220, 262, 1, 196, 247, 1, 220, 262, 4, 165, 196 3640 DATA .65,196,247,.5,220 ,262,1,196,247,1,165,196,1,2 62,330,1,220,262,3,196,247 3650 DATA .5,196,247,.5,220, 262, .5, 196, 247, .5, 220, 262, 1, 196,247,1,262,330,4,247,294 3660 DATA .65,175,220,.5,196 ,247,1,173,220,1,147,175,1,2 47,294,1,220,262,3,196,247 3670 DATA .5,196,247,.5,220, 262, .5, 196, 247, .5, 220, 262, 1, 196,247,1,294,349,4,262,330 3680 DATA 1,220,262,1,220,26 2,1,262,330,1,220,262,1,196, 247, 1, 165, 196, 2, 196, 247 3690 DATA 1,175,220,1,220,26 2,1,196,247,1,175,220,4,165, 196 3700 DATA 1,147,175,1,165,19 6,1,196,247,1,220,262,1,247, 294,1,247,294,2,247,294 3710 DATA 1,262,330,1,262,33 0,1,247,294,1,220,262,1,196, 247,1,175,220,2,147,175

3720 DATA .65,196,247,.5,220 ,262,1,196,247,1,165,196,1,2 62,330,1,220,262,3,196,247 3730 DATA .5,196,247,.5,220, 262, .5, 196, 247, .5, 220, 262, 1, 196,247,1,262,330,4,247,294 3740 DATA .75,175,220,.7,220 ,262,1,175,220,1,147,175,1,2 47,294,1,220,262,3,196,247 3750 DATA .5,196,247,.5,220, 262, .5, 196, 247, .5, 220, 262, 1, 196,247,1,294,349,3,262,330, 1,20000,20000 4000 M=450 !FIRST NOEL 4010 RESTORE 4400 :: GOSUB 6 000 :: X=7 :: RESTORE 4600 : : GOSUB 7000 4020 RESTORE 4410 :: GOSUB 6 000 :: X=6 :: RESTORE 4610 : : GOSUB 7000 4030 RESTORE 4420 :: GOSUB 6 000 :: X=7 :: RESTORE 4620 : : GOSUB 7000 4040 RESTORE 4430 :: GOSUB 6 000 :: X=5 :: RESTORE 4630 : : GOSUB 7000 4050 RESTORE 4440 :: GOSUB 6 000 :: X=7 :: RESTORE 4600 : : GOSUB 7000 4060 RESTORE 4450 :: GOSUB 6 000 :: X=6 :: RESTORE 4610 : : GOSUB 7000 4070 RESTORE 4460 :: GOSUB 6 000 :: X=7 :: RESTORE 4620 : : GOSUB 7000 4080 RESTORE 4470 :: GOSUB 6 000 :: X=5 :: RESTORE 4630 : : GOSUB 7000 4090 RESTORE 4480 :: GOSUB 6 000 :: X≃7 :: RESTORE 4600 : : GOSUB 7000 4100 RESTORE 4490 :: GOSUB 6 000 :: X=5 :: RESTORE 4640 : : GOSUB 7000 4110 RESTORE 4500 :: GOSUB 6 000 :: X=5 :: RESTORE 4650 : : GOSUB 7000 4120 RESTORE 4510 :: GOSUB 6 000 :: X=5 :: RESTORE 4660 : : GOSUB 7000 :: RETURN 4400 DATA THE__ FIRST__ NO-E L,O 4410 DATA THE__ AN-GEL DID S AY,2 4420 DATA WAS TO CER-TAIN PO OR SHEP-HEARDS, 2 4430 DATA IN FIELDS AS THEY LAY;,3

4440 DATA IN__ FIELDS__ WHER E___ THEY, 2 4450 DATA LAY__ KEEP-ING THE IR SHEEP, 2 4460 DATA ON A COLD WIN-TER(S NIGHT___,2 4470 DATA THAT WAS__ SO DEEP , 2 4480 DATA NO-EL__ NO-EL NO-E L,2 4490 DATA NO-EL NO-EL,2 4500 DATA BORN IS THE KING___ 4510 DATA OF IS-RA-EL.,2 4400 DATA .5,165,165,.5,147, 147,1,131,131,.5,147,147,.5, 165, 165, .5, 175, 175, 2, 196, 196 4610 DATA .5,220,262,.5,247, 294,1,262,330,1,247,294,1,22 0,262,2,196,247 4620 DATA .5,220,262,.5,247, 294,1,262,330,1,247,294,1,22 0,262,1,196,247,1.5,220,262 4630 DATA 1,247,294,1,262,33 0,1,196,247,1,175,220,2,165, 196 4640 DATA .5,262,330,.5,247, 294,2,220,262,1,220,262,3,19 6,247 4650 DATA 1,262,330,1,247,29 4,1,220,262,1,196,247,1.5,22 0,262 4660 DATA 1,247,294,1,262,33 0,1,196,247,1,175,220,2,165, 196 5000 M=300 !JINGLE BELLS 5010 RESTORE 5400 :: GDSUB 6 000 :: X=5 :: RESTORE 5600 : : GOSUB 7000 5020 RESTORE 5410 :: GOSUB 6 000 :: X=7 :: RESTORE 5610 : : GOSUB 7000 5030 RESTORE 5420 :: GOSUB 6 000 :: X=5 :: RESTORE 5620 : : GOSUB 7000 5040 RESTORE 5430 :: GOSUB 6 000 :: X=5 :: RESTORE 5630 : : GOSUB 7000 5050 RESTORE 5440 :: GOSUB 6 000 :: X=5 :: RESTORE 5640 : : GOSUB 7000 5060 RESTORE 5450 :: GOSUB 6 000 :: X=5 :: RESTORE 5650 : : GOSUB 7000 5070 RESTORE 5460 :: GOSUB 6 000 :: X=8 :: RESTORE 5660 : : GOSUB 7000

5080 RESTORE 5470 :: GOSUB 6 000 :: X=7 :: RESTORE 5670 : : GOSUB 7000 5090 RESTORE 5480 :: GOSUB 6 000 :: X=6 :: RESTORE 5680 : : GOSUB 7000 5100 RESTORE 5490 :: GOSUB 6 000 :: X=5 :: RESTORE 5690 : : GOSUB 7000 5110 RESTORE 5500 :: GOSUB 6 000 :: X=7 :: RESTORE 5700 : : GOSUB 7000 5120 RESTORE 5510 :: GOSUB 6 000 :: X=8 :: RESTORE 5710 : : GOSUB 7000 :: CALL CLEAR 5130 RESTORE 5480 :: GOSUB 6 000 :: X=6 :: RESTORE 5680 : : GOSUB 7000 5140 RESTORE 5490 :: GOSUB 6 000 ;: X=5 ;; RESTORE 5690 ; : GOSUB 7000 5150 RESTORE 5500 :: GOSUB 4 000 :: X=7 :: RESTORE 5700 : : GOSUB 7000 5160 RESTORE 5510 :: GOSUB 6 000 :: X=8 :: RESTORE 5720 : : GOSUB 7000 :: RETURN 5400 DATA DASH-ING THRU THE SNOW, O 5410 DATA IN A ONE HORSE O-P EN SLEIGH, 2 5420 DATA CIER THE FIELDS WE GO, 2 5430 DATA LAUGH-ING ALL THE WAY:,2 5440 DATA BELLS ON BOB-TAIL RING, 2 5450 DATA MAK-ING SPIR-ITS B RIGHT, 2 5460 DATA WHAT FUN IT IS TO RIDE AND SING, 2 5470 DATA A SLEIGH-ING SONG TO-NIGHT, 3 5480 DATA JIN-GLE BELLS JIN-GLE BELLS,2 5490 DATA JIN-GLE ALL THE WA Y,2 5500 DATA OH WHAT FUN IT IS TO RIDE, 2 5510 DATA IN A ONE HORSE O-P EN SLEIGH, 2 5600 DATA 1,196,247,1,330,39 2,1,294,349,1,262,330,3,196, 247 3610 DATA .5,196,247,.5,196, 247,1,196,247,1,330,392,1,29 4,349,1,262,530,4,220,262

5620 DATA 1,220,262,1,349,44 0,1,330,392,1,294,349,4,247, 294 5630 DATA 1,392,494,1,392,49 4,1,349,440,1,294,349,4,330, 392 5640 DATA 1,196,247,1,330,39 2,1,294,349,1,262,330,4,196, 247 5650 DATA 1,196,247,1,330,39 2,1,294,349,1,262,330,3,220, 262 5660 DATA 1,220,262,1,220,26 2,1,349,440,1,330,392,1,294, 349,1,392,494,1,392,494,2,39 2,494 5670 DATA 1,392,494,1,440,52 3,1,392,494,1,349,440,1,294. 349,3,262,330,1,20000,20000 5480 DATA 1,330,392,1,330,39 2,2,330,392,1,330,392,1,330, 392,3,330,392 5690 DATA 1,330,392,1,392,49 4,1,262,330,1,294,349,4,330, 392 5700 DATA 1,349,440,1,349,44 0,1,349,440,1,349,440,1,349, 440,1,330,392,2,330,392 5710 DATA .5,330,392,.5,330, 392,1,330,392,1,294,349,1,29 4,349,1,330,392,2,294,349,2, 392,494 5720 DATA .5,330,392,.5,330, 392,1,392,494,1,392,494,1,34 9,440,1,294,349,3,262,330,1, 20000,20000 6000 READ L#,L :: Y=Y+L :: D ISPLAY AT(1+Y,1):L# :: RETUR 7000 FOR BT-1 TO X :: READ D ,T1,T2 :: CALL SOUND (M*D,T1, 3,T2,5,T1*2,9):: NEXT BT :: RETURN 10000 CALL SCREEN(15):: CALL CLEAR :: CALL CHAR(96, "0003 OFOF3F3F3F7=",97,"7E70C30707 OF1F06",98, "FFFEFEF8F8E080", 104, *CCCCFFFCCCCFF3F3*) 10010 CALL COLOR(9,3,16,10,4 ,2) 10020 RA=1 :: RB=2 :: FOR CA =1 TO 28 STEP 2 :: CB=CA :: GOSUB 11000 :: NEXT CA 10030 RA=23 :: RB=24 :: FOR CA=1 TO 28 STEP 2 :: CB=CA : : GOSUB 11000 :: NEXT CA 10040 CA.CB=1 :: FOR RA=3 TO 21 STEP 2 :: RB=RA+1 :: GOS UB 11000 :: NEXT RA 10050 CA,CB=27 :: FOR RA=3 T 0 21 STEP 2 :: RB=RA+1 :! GO SUB 11000 :: NEXT RA 10060 CALL COLOR(0,10,16):: FOR A=2 TO 8 :: CALL COLOR(A ,10,16):: NEXT A :: CALL CHA R(40, "08081"):: CALL CHARPAT (35, CH\$):: CALL CHAR(41, CH\$) :: RETURN 11000 DISPLAY AT (RA, CA) SIZE (2):"'";"h" :: DISPLAY AT(RB, CB) SIZE(2): "a"; "b" :: RETURN 12000 FOR X=4 TO 10 STEP 2 : : READ COL,G\$:: DISPLAY AT(X,COL)SIZE(LEN(G\$)):G\$:: NE XT X :: RETURN 12010 DATA 2, MERRY, 1, CHRISTM AS,4,TO,3,ALL,21,AND A,20,HA PPY, 22, NEW, 23, YEAR





HU99 L I BRARY NEWS HTIW AL LANRENCE



HI 99'ers. ****Adventurers***

HINTS-PIRATE

- a. Enter crack and explore shed, all.
- b.Pull nails somewhere.
- c.Doubloons are pieces of eight. Who 9.Play around with Thrust and Pitc have you heard talking about pieces to Trim for level Flight and tr of eight lately.
- d.Try 30 paces in field and dig.

ADVENTURELAND

a. Stuck in a pit? say a magic word landing and that while carrying something Brick wall a problem? throw the at it b. Wasting Honey? leave mirror in the the area as I know of a Spitfire Ac stump, its magic c.LAVA a problem mirror is good for Skys. (GRAPHICS in this one) many clues, even sleeping Dragons! d.Does the hear have a Crown?

Need any more clues get in touch with See you in the New Year Rodney Gainsford at meetings Ph.583515

We received 2 new PUBLIC DOMAIN disk from our friends in U.K. to add t our library, some of these are o this months cassette they are available as disk lots so if you would like these let me know.

Library

Any clubs or indivuals interested obtaining any PUBLIC DOMAIN softwar in volume disks have 2 choices

- (a) Send blank initialised disks us with return postage or send u disks with programs on it and we wil send at our cost an equal number of disks filled with programs requeste or volume disks.
- (b) We can supply programs, or volume disks on our disks for the cost disk and PP(\$4.00)

Flight Simulator(Ex.Basic) that! on instruments only you expected to take off climb selected heights on new compas headings each time. Some tips for Non Biggles type air Aces

- 1. Set compass heading.
- 2.Select full Flaps.
- 3. Pitch Neg.
- 4. Thrust to Max.
- 5.Release Brake.
- 6. When speed 150K+ select Pitch to and you should be airborne.
- take 7. Gear up flaps to Min.
 - 8. Between ist. and 2nd. Altitude Sel 2nd. Heading and Up Up and away!!.
 - aerobatics.
 - 10. Had enough Press O and you have heading of Runway displayed on the panel.
 - 11.Pray as this is your First SOLE Ground is HARD!!!

Ps.Anyone know of a BBC computer willing to shoot it out in thos

Merry Christmas. Dont Drink+Fly

Al Lawrence.

disks d to re on also f you

d in tware

s to nd us will er of ested

olume t of

just are nb to mpass as for

to ·

de Set y!!. Pitch nd try

have on the

SOLO still

er in re Ace those

DO IT YOURSELF THREE IN ONE MODULE

MAKING ONE MODULE THAT COMBINES E/A DISK MANAGER AND TI-WRITER.

If you've ever opened any of your command modules like the ones listed above, you'll see the infamous GROM The modules that use these chips. GROMs have corresponding pins on all The GROM chips bussed together. itself determines the addressing and there can be one or more GROMs depending on the cartridge. There is a common reset line (pin 14) all the chips and MULTIPLAN uses 5 of these GROMS, all in parallel. One thing all these cartridges have in common is the circuit board that has room for up to 5 GROMS.

If you were to take the TI-WRITER module and E/A that both have only one GROM each, and put both these GROMs on one board, option 1, BASIC would be O.K.. However, option 2 would be garbage because each GROM is trying to display its function. To get around this problem you have

to use separate resets to each of the GROM sets. To do this I removed solder from all the blank i.c. locations on the TI-WRITER board (or any board you'll be using that has one GROM) and removed the 100 resistor which you'll reuse later. You now have to cut the reset (the run connecting pin 14 on all chips) between the chip sets for the GROMs you are adding, i.e. DISK MANAGER II has 2 GROMs so cut run on either side of where you insert those chips, but not between.

to add a switch to You now have connect the 100 Ohm resistor to each of the GROM sets you are using. I'll leave it up to you what you use the switch. If you only have two functions in the cartridge, then position miniature slide switch is ideal. If you have three you could use a DIP functions. switch but then you have to remember to only have 1 switch on at a time or you'll get garbage when you try to make a 2nd or 3rd selection. 100 Ohm resistor has to be connected from the common on the switch to the hole on the outside edge of the board where it was originally. switch can be located directly over the location the resistor originally occupied and you will have to cut the top of the cartridge in you have access to the ____ 50 switch slide or whatever.

important to remember that any modification is your responsibility if anything goes wrong, you are the one to suffer, not me. Try make sure that everything will physically fit and be VERY careful rmoving and handling i.c's. recommend using sockets on the board you are going to use (clean the i.c. pins of all solder before inserting sockets) so you can in replace in the original them cartridge (use sockets here also) if you decide to undo the modification Some of the cheaper for any reason. modules like MunchMan have 4 or 5 possible i.c. locations. If you use the MunchMan cartridge, you can also include the 8K ram modification that Ron Gries and John Clulow came up with and have a really super cartridge!!

Richard J. Bailey NH99ER USER GROUP (3

 $(^{\circ})$

M

M

B

0

Q

CONNED

Not so long ago my wife returned home from one of her professional meetings with the lovely words "Guess what?".

Now I don't know what you might think when you hear this sort of thing from your spouse, but my legs trembled in anticipation(really), and I dutifully asked "What?". (all the time fearing the worst and boy did I get it.)

"That computer of yours... It can print mailing labels ,can't it ?(asked rhetorically). I told the ladies that it could and that you would do the labels for the secretary... That's alright, isn't it?"

Well, what would you do. I had to agree (as if she didn't know.. after spending all that money what else could I do.)

Well.. to cut a long story short I looked through my programs and the closest thing I could find telephone/address wa q sorter/storer program called *HOMSECTRY* (I'd say that most of you have a copy of well this structured piece of basic programming) and I set about bastardising it.

The final product(now X/B) lets you construct files on up to 100 people using the following prompts...

SURNAME.

C/NAME (include title)

TELEPHONE NO.

STREET

CITYLPOSTCODE

MISCELLANEOUS (eg 'F' financial 'U' unfinancial and any other comments you wish to add.)

The program will then sort the data according to the surname and allow you to many things with it.

A menu allows you to store; alter; update; display all items; display one item; or print data.

Since this last option was the one I wanted most, the program allows you to print the data a number of ways.

You can list ALL MEMBERS AND ALL INFORMATION or ALL MEMBERS AND

MAILING INFORMATION OF ALL FINANCIAL MEMBERS AND ALL INFORMATION OF ALL FINANCIAL MEMBERS AND MAILING INFORMATION.

(Note.. when printed, the title and given name appear before the surname.)

Double or single strike printing can be employed and the line spacing can be varied by altering a variable in the program (clumsy I know but I made it only for me so I didn't think to make it easily varied.)

The program will only print single width rolls with a label size of 87mm by 23mm working well.

Anyhow ... it worked and now my computer and I are a hit with the ladies. (And boy is that good 'cause they are all home science teachers and send me tasty little morsels in return for the labels.)

P.S. The program is now available to club members from our hard working librarian and with apologies to the original creator for my corruption of his well ordered code.

Merry Christmas. Peter Smith



lets 100 wing

ncial other

> the and

tore; tems;

the ogram ta a

list N or

ALL

NCIAL .

title the

trike the

d by ogram only

it

print size

d now h the cause

chers ls in

now m our with eator well

mith

"HORSEPLAY"





THIS PROGRAMME IS NOT A GAME BUT A PROGRAMME TO WORK OUT RACING ODDS ON THE GEE GEES. I THINK IT IS RUNNING ALLRIGHT, BUT NOT BEING A RACING MAN I AM NOT FULLY SURE ABOUT THIS POINT. IF ANY OF YOU OUT THERE CAN GIVE ME ANY CONSTRUCTIVE ADVICE ON THE PROGRAMME, PLEASE LET ME KNOW.

LASTLY AS I HAVE CONVERTED THE PROGRAMME FROM AN ATARI PROGRAMME I AM NOT SURE THAT I HAVE GOT IT COMPLETELY RIGHT. SO PLEASE DON'T BLAME ME IF YOU DO YOUR DOUGH, BUT IF YOU START TO DO ALLRIGHT OUT OF IT DON'T FORGET WHO CONVERTED THE PROGRAMME, AS ALL DONATIONS WILL BE GRATEFULLY ACCEPTED.

BRIAN R.

210 DISPLAY ERASE ALL :: CAL L SCREEN(5):: FOR I=0 TO 12 :: CALL COLOR(I,16,1):: NEXT

220 OPTION BASE 1 :: CALL CH AR(95, "OOFFOOFF"):: DIM A\$(1 2),H(12),ES(12),WT(12),SO(12

230 CALL MEN :: CALL CH(6,K) 240 ON K-48 GOSUB 260,270,28 0,290,300,310

260 CALL HDC(A\$(),H(),TRK\$,D ATE\$,RACE\$,N):: RETURN

250 GOTO 230

270 CALL RF(A\$(),H(),TRK\$,DA TE\$,RACE\$,N):: RETURN

280 CALL SF(A\$(),H(),TRK\$,DA TE\$,RACE\$,N):: RETURN

290 CALL SCR(As(),H(),N,TRKs,DATEs,RACEs):: RETURN

300 CALL PR(A\$(),H(),TRK\$,DA TE\$,RACE\$,N):: RETURN 310 DISPLAY ERASE ALL :: STO

320 SUB CH(X,K):: DISPLAY AT (23,2)BEEP: "Your choice"

330 CALL KEY(3,K,S):: IF S<1
THEN 330 ELSE IF K>48 AND K
<(X+49)THEN 350
340 DISPLAY AT(24,1)BEEP: "A
number between 1 and ";X :: G
OTO 330
350 CALL CUR(23):: SUBEND

360 SUB MEN :: DISPLAY AT(1, 10)ERASE ALL: "Horse play" :: DISPLAY AT(2,9):RPT\$("_",12

370 DISPLAY AF(4,6):"1 Hdcp a new race":"":" 2 Get s aved race":"":" 3 Save t his race":"":" 4 Scratch a horse"

380 DISPLAY AT(12,6):"5 Print this race":"":" 6 Exit

390 SUBEND

400 SUB CUR(R):: FOR I=24 TO R STEP -1 :: CALL HCHAR(I,1,32,32):: NEXT I :: SUBEND

410 SUB SAC(R,C,B\$)
420 ACCEPT AT(R,C)VALIDATE(D
IGIT)BEEP:B\$:: IF B\$="" THE
N 420
430 SUBEND

440 SUB CM(K):: DISPLAY AT(2 4,1): "Change mind Y/N" :: CA LL YN(K):: DISPLAY AT(24,1): "" :: SUBEND

450 SUB YN(K)

470 SUBEND 480 SUB HDC(A\$(),H(),TRK\$,DA TE\$, RACE\$, N):: CALL CUR(3) 490 DISPLAY AT(4,1): "Which t rack.....":"Race date..... ..":"Date 30 days ago.":"Whi ch race.....": "How many ho rses.." 500 N=18 :: ACCEPT AT(4,N)SI ZE(-10)BEEP:TRK\$:: ACCEPT A T(5,N)BEEP SIZE(-8):DATE# :: ACCEPT AT(6,N)BEEP SIZE(-8) :DAYS :: ACCEPT AT(7,N)BEEP SIZE(-2):RACE# 510 CALL SAC(8,N,B\$):: N=VAL (B\$):: IF N<1 OR N>12 THEN N =18 :: GOTO 510 :: ELSE CALL CM(K):: IF K=89 THEN 500 520 DISPLAY AT(10,1):"Is thi s race a....":" 1 Claim ing race":" 2 Allowance race":" 3 Stakes race" 4 Handicap race* 530 CALL CH(4,K):: CLASS=K-4 8 :: DISPLAY AT(16,1):"One m Y/N" :: CALL ile or more YN(FL):: DISPLAY AT(18,1):" Is this a grass race Y/N" :: CALL YN (GR) 540 CALL CM(K):: IF K=89 THE N CALL CUR(10):: GOTO 520 :: ELSE CALL RD(TRK\$, DATE\$, RAC E\$):: DISPLAY AT(6,1)BEEP:"T ype in the horses names" 550 CALL HD(A\$(),N):: FOR I= 1 TO N :: ACCEPT AT(I+8,16)\$ IZE(-12):A\$(I):: NEXT I :: C ALL CM(K) 560 IF K=89 THEN 550 570 FOR I=1 TO N :: H(I)=10 :: NEXT I :: CALL CUR(5):: D ISPLAY AT(6,1):"Lets figure. 580 FOR I=1 TO N :: CALL CUR (7):: CALL SCREEN(14):: DISP LAY AT(6,16) BEEP: A\$(I):: FOR J=1 TO 500 :: NEXT J :: CAL L SCREEN(5) 590 ON CLASS GOSUB 790,810,8 60,860 600 CALL CUR(7):: DISPLAY AT (8,1)BEEP: "Has he raced sinc e ";DAY\$;" Y/N" :: CALL YN(K

):: IF K=78 THEN FP=0 :: GOT

0 620

460 CALL KEY(3,K,S):: IF S(1

OR(K<>78 AND K<>89)THEN 460

610 DISPLAY AT(10,1) BEEP: "En ter position last race 2 3 etc." :: CALL SAC(11,12, B\$):: FP=VAL(B\$) 620 DISPLAY AT(13,1)BEEP: "In the last 6 races at a jor track." :: DISPLAY AT(16 ,6): "How many wins" :: CALL SAC(16,23,8\$):: WINS=VAL(B\$) 630 DISPLAY AT(17,6)BEEP: "Ho w many seconds" :: CALL SAC(17,23,B\$):: TWOS=VAL(B\$) 640 DISPLAY AT(19,1) BEEP: "At a minor track":"":" many wins" :: CALL SAC(21,2 3,B\$):: WONS=VAL(B\$):: CALL CM(K):: IF K=89 THEN 600 450 H(I)=H(I)+(-5*(FP=1))+(-7*(FP=2))+(-2*(FP=3))+(1*(FP =5))+(2*(FP=6))+(4*(FP>6)):: CPTS=WINS*2+TWOS+WONS 660 H(I)=H(I)+(-8*(CPTS>=6)) +(-6*(CPTS=5))+(-4*(CPTS=4)) +(-3*(CPTS=3))+(-1*(CPTS=2)) +(2*(CPTS=1))+(4*(CPTS=0)) 670 CALL CUR(7):: DISPLAY AT (8,1)BEEP: "Find and add the two bests first call positi ons at thisdistance or short er.":"":"For example..." 680 DISPLAY AT(13,9):"(5) 4 4 1 PINCAY":" (+2) 2 1 1 SHOEMKR":" ~----7 Enter here->" :: D ISPLAY AT(17,4): "Enter today weight in 1bs." 690 CALL SAC(16,25,B\$):: ES(I)=VAL(B\$):: CALL SAC(18,25, B\$);; WT(I)=VAL(B\$);; CALL C M(K):: IF K=89 THEN 690 700 IF FL=89 THEN GOSUB 890 ELSE GOSUB 910 710 NEXT I 720 CALL CUR(5):: DISPLAY AT (10,1): "Sorting out the odds for you" :: FOR I=1 TO N :: SORT(I)=ES(I):: NEXT I :: G OSUB 930 730 FOR I=1 TO N :: IF ES(I) <=SORT(3)THEN H(I)=H(I)+3 EL SE H(I)=H(I)-2 740 NEXT I 750 FOR I=1 TO N :: SORT(I)= WT(I):: NEXT I :: GOSUB 930 760 FOR I=1 TO N :: IF WT(I) >=SORT(N-2) THEN H(1)=H(1)+4 ELSE H(I)=H(I)-3770 IF H(I) (=0 THEN H(I)=1

780 NEXT I :: SUBEXIT 790 DISPLAY AT(8,4)BEEP: "Has he ever finished": " 1st u r 2nd in this":" class or higher Y/N* 800 CALL YN(K):: IF K=89 THE N H(I)=H(I)+4 :: RETURN :: E LSE H(I)=H(I)-3 :: RETURN 810 DISPLAY AT(8,1) BEEP: "Has he ever raced in a":"stakes or handicap race Y/N" 820 CALL YN(K):: IF K=78 THE N 840 ELSE DISPLAY AT(12,4)B EEP: "Has he ever run in a":" claimer since Y/N" :: CAL L YN(K) 930 IF K=89 THEN H(I)=H(I)-4 :: RETURN :: ELSE H(I) = H(I) +7 :: RETURN 840 DISPLAY AT(12,4)BEEP: "Ha s he only run": " allowance or non";" claiming maiden race Y/Nº :: CALL YN(K):: I F K=78 THEN H(I)=H(I)-4 850 RETURN 860 DISPLAY AT(8,1) BEEP: "Has he ever raced in a sta kes or handicap race Y/N" :: CALL YN(K) 870 IF K=89 THEN 880 ELSE H(I)=H(I)-4 :: RETURN 880 DISPLAY AT(11,1)BEEP: "Di d he win Y/N" :: CALL YN(K): : IF K=78 THEN RETURN :: ELS E H(I)=H(I)+6 :: RETURN 890 CALL CUR(7):: DISPLAY AT (8,4)BEEP: "Was his last race a mile or more Y/N" :: CALL YN(K):: IF K=89 THEN H (I)=H(I)+3 :: RETURN 900 DISPLAY AT(10,1) BEEP: "An y race a mile or more Y/N" : : CALL YN(K):: IF K=78 THEN RETURN :: ELSE H(I)=H(I)+1 : : RETURN 910 CALL CUR(7):: DISPLAY AT (8,4) BEEP: "Was his last race on":" the turf Y/N" :: CA LL YN(K):: IF K=89 THEN H(I) =H(I)+3 :: RETURN 920 DISPLAY AT(10,4)BEEP: "An y race on the":" the turf Y/N" :: CALL YN(K):: IF K=78 THEN RETURN :: ELSE H(I)=H(I)+1 :: RETURN 930 M=N 940 M=INT(M/2):: IF M=0 THEN RETURN

950 FOR J=1 TO N-M :: I=J 960 L=I+M 970 IF SORT(I) <= SORT(L) THEN 990 980 X=SORT(I):: SORT(I)=SORT (L):: SORT(L)=X :: I=I-M :: IF I>=1 THEN 960 990 NEXT J :: GOTO 940 1000 SUBEND 1010 SUB CD(B\$):: CALL CUR(5):: DISPLAY AT(6,1):B\$:: DI SPLAY AT(8,1):"1 CASSETTE" : : DISPLAY AT(10,1):"2 DISK" :: CALL CH(2,K) 1020 IF K=49 THEN DISPLAY AT (8,14): "CS1" ELSE DISPLAY AT (10,14): "DSK1.RACE/ODDS" 1030 ACCEPT AT((-8*(K=49))+(-10*(K=50)),14)SIZE(-15):B\$
:: IF B\$="" THEN 1030 1040 SUBEND 1050 SUB SF(A\$(),H(),TRK\$,DA TE\$, RACE\$, N) 1060 B\$="Saving to...." :: C ALL CD(B\$):: DISPLAY ERASE A LL :: OPEN #1:B\$, INTERNAL, OU TPUT, FIXED 192 1070 PRINT #1:TRK\$, DATE\$, RAC E\$,N :: FOR I=1 TO N-1 :: PR INT #1:A\$(I),STR\$(H(I)),::NEXT I :: PRINT #1:A\$(N),STR\$ (H(N)):: CLOSE #1 1080 SUBEND 1090 SUB RF(A\$(),H(),TRK\$,DA TE\$, RACE\$, N) 1100 B\$="Reading from..." : : CALL CD(B\$):: DISPLAY ERAS E ALL :: OPEN #1:B\$, INTERNAL ,INPUT ,FIXED 192 1110 INPUT #1:TRK#, DATE#, RAC E\$,N :: FOR I=1 TO N-1 :: IN PUT #1:A\$(I),B\$,:: H(I)=VAL(B\$):: NEXT I :: INPUT #1:A\$(N), B\$:: H(N)=VAL(B\$):: CLOS E #1 1120 SUBEND 1130 SUB RD(TRK\$, DATE\$, RACE\$):: CALL CUR(3):: DISPLAY AT (4,1):TRK#;" ";DATE#;" RACE "; RACES :: SUBEND

1140 SUB HD(A\$(),N):: FOR I= 1 TO N :: DISPLAY AT(I+8,1): "No.";I;TAB(8);"horse...";A\$ (I):: NEXT I :: SUBEND

1150 SUB KC :: DISPLAY AT(24,1)BEEP: "Press any key to continue"
1160 CALL KEY(3,K,S):: IF S

1 THEN 1160

1170 SUBEND

1180 SUB SCR(A\$(),H(),N,TRK\$,DATES,RACES):: CALL RD(TRKS ,DATE\$,RACE\$):: CALL HD(A\$() , N) 1190 DISPLAY AT(21,1) BEEP: "T ype in the horse No.": "or ze ro to exit* 1200 CALL SAC(22,24,B\$):: I= VAL(B\$):: IF I=0 THEN SUBEXT T :: ELSE IF I>N THEN 1190 1210 CALL CUR(21):: DISPLAY AT(21,1)BEEP: "This horse? Y/ N ";A#(I);; CALL YN(K);; IF K≠78 THEN 1190 1220 FOR K=I TO N-1 :: A\$(K) =A\$(K+1):: H(K)=H(K+1):: NEXT K :: A\$(N)="" :: H(N)=0 :: N=N-1 :: CALL CUR(6):: CALL HD(A\$(),N) 1230 DISPLAY AT(22,1) BEEF: "T he field as it now stands" :

1240 SUB PR(A\$(),H(),TRK\$,DA TE\$,RACE\$,N):: CALL CUR(3)
1250 DISPLAY AT(6,4):"Print to...." :: DISPLAY AT(8,9):"
1 Screen" :: DISPLAY AT(10,9):"2 Printer" :: CALL CH(2,K):: K=K-49 :: IF K=O THEN DI SPLAY ERASE ALL :: GOTO 1280

: CALL KC :: SUBEND

1260 DISPLAY AT(10,19): "PIO" :: ACCEPT AT(10,19)BEEP SIZ E(-3):B\$:: IF B\$="" THEN 12 40 ELSE CALL CUR(5):: DISPLA Y AT(12,9): "Printing to ";B\$ 1270 OPEN #K:8\$ 1280 PRINT #K: TAB(9+(K-1)*-2 7); "Horse Play": TAB(8+(K=1)* -27);RPT\$("~",11):: PRINT #K 1290 PRINT #K: TAB(1+(K=1) *-1 0); "Race "; RACE\$; TAB(9+(K=1) *-18); TRK\$; TAB(20+(K=1)*-30) : DATES 1300 PRINT #K:TAB(1+(K=1)*-2 6);RPT\$("~",28+(K=1)*-47) 1310 FOR I=1 TO N :: TOT=TOT +H(I):: NEXT I :: TOT=TOT*.7 9 :: NU=.18 :: FOR I=1 TO N :: PCT=H(I)/TOT 1320 IF PCT=0 THEN ODDSs="OU T* :: GOTO 1350 :: ELSE IF P CT<NU THEN ODDS#=STR#(INT(1/ PCT-1)):: GOTO 1350 1330 FOR J=1 TO 17 :: READ T BL, ODDS# :: IF PCT>TBL THEN 1350 1340 NEXT J 1350 PRINT #K:TAB(5+(K=1) #-1 6);A\$(I);TAB(20+(K=1)*-30);Q DDS\$:: RESTORE 1370 :: NEXT I :: PRINT #K :: PRINT #K 1360 IF K=1 THEN CLOSE #K :: SUBEXIT :: ELSE CALL KC :: SUBEXIT 1370 DATA .85,1-5,.71,2-5,.6 7,1-2,.62,3-5,.55,4-5,.5,EVE N,.45,6-5,.41,7-5,.4,3-2,.38 ,8-5,.35,9-5 1380 DATA .33,2,.28,5-2,.25, 3,.22,7-2,.2,4,.18,9-2 1390 SUBEND

*)

I

ar

wł

gi

dr

12

MC

re

al

• 9

Ja

an

CO

25

be

to

Th

st

dr

15

рl

i m

f i

no

ma

th

開幕

is fu

The er: su| it

YOU

dis

the

of wo:

ba:

not

las

501

(Reput

WISH LIST

LASER REPAIRED TECHNOLOGY

In 1978 Philips were the first to announce a laser based data drive which could hold 1000 megabytes (1 gigabyte). Research into laser drives has been going on for 12 years and usually as a result of work done in the field of recorders. Currently most, if not all, manufacturers of equiptment, in America, Europe and Japan, are in the process announcing laser drives and some computer manufacturers as well, such as Burroughs and STC. Capacities being offered are in the range of 1 to 4 gigabytes.

The great advantage, apart from vast storage capacities, of the laser drive is its robustness. The data is burned onto the surface of the platter and consequently is almost immune to damage. Such things as fingers, spilt coffee and dust have no effect on laser disks. This makes them particularly suited to the smaller end of the computer market where, typically the computer is treated rather like a piece of furniture.

The main disadvantage is the non erasibility of the data. Some supporters of laser disks say that it is not important to erase data; you simply take another part of the disk to update your data and ignore the previous entry. With a couple of gigabytes to play with, there would be enough storage to last the smaller user for some years on this Manufacturers are generally not satisfied with this idea and laser disks will have erasibility by sometime in the late 1980s. (Research currently being carried out indicates that by using a laser beam of lower intensity to remelt the the disk surface to erase data. disks may be re-used as is the case with floppies).

Laser disks do not have the same dificulties with head alignment as magnetic disks and are removable as standard, which relieves any problems concerning backing up data.

Remarkably the indicated cost of a 1 to 2 gigabyte laser disk is less than that of a 5 1/4 inch floppy with 2 megabytes. Even assuming that floppy disk drives come down drastically in price, there is obviously no way they can match the price per bit of a laser disk. It is quite possible that laser disks will eventually replace all other types of disks.

TIII BLAGII IIOLI

You wouldn't believe it after filling the rest of the magazine with heaps of articles I'm left with this pokey little hole to fill. I could probably fill it up with some fancy graphics or some hints and tips but I'm afraid I've just

plain run out of them. So I guess I will just waffle away until the hole is filled. One thing I should mention is the Christmas get together for all the regular contributors to the HV99 NEWS. starts at 8PM on Christmas eve and will be held in the phone box on Cherry Road Warners Bay. (Very Nasty) But seriously though in this issue a number of new people have started to contribute to magazine this is and feel reflected quality in it≂ diversity. What I'm getting at is if you intend to make any new years resolutions next year put the magazine high on your list priorities.

A merry Christmes to all and I look forward to sesing you all again in the New Year.

Steve

ON CHIENES INM

SYSTEM CONFIGURATION CHART

R = Required O = Optional	THE PLAYERACE	Pares Ports	CECH SYNTHE	TAKER	EXTEMDED TEIL	PERION SYSIC	WEWSON CHA	HIENE OF CA	600 P.COOC	DISK DE CARD	PAINE AD	PHONE W	OFCIAL CARTE	mare.
CASSETTE STORAGE/RETRIEVAL		R												
DATA BASE ACCESS				0	R		R		R		0	0	R	TEH
DISK STORAGE/RETRIEVAL							Ħ				R			Disk Manager
GRAPHICS (FI BASIC)														
MUSIC	R									L				
SPEECH	R			R	0	0								Yes*
PRINTING	R						Ŕ		R			R		
ASSEMBLY (EDITOR/ASSEMBLER)	R					0	Ř	Я	0		R	0		Yes
ASSEMBLY (Mint Memory)	A	R			l		0	٥						Yes'"
UCSD PASCALTM‡	Я	0		0			Я	R	0	R.	0	0	Γ.	Yes***
TI-PILOT	Ř			0			R	R	0	R	Ħ	0		Yes****
TI-BASIC	A A	٥		٥			۵	0	0		٥	0		. • •
TI-EXTENDED BASIC		0		0		А	0	٥	0		0	0		Yes
TI-LOGO		0			:		R,	R			0	ļ	i	Yes++
CASSETTE SOFTWARE	А	P												L
DISK SOFTWARE	R R			L		<u></u>	R				R			
EDUCATIONAL CARTRIDGES 1, 2			0	0										Yes
GAME CARTRIDGES ¹			0	0										Yes
HOUSEHOLD BUDGET MANAGEMENT		٥		<u> </u>		ļ	¢		0		٥	٥		Yes
HOME FINANCIAL DECISIONS		<u> </u>	<u> </u>	<u>. </u>					L		L			Yes
MICROSOFT TM MULTIPLAN (Spread Sheet) ³							R	R	0		R	0		Yes
PERSONAL REAL ESTATE		0		<u> </u>			0		0			0		Yes
PERSONAL RECORD KEEPING		0	<u> </u>			ļ	٥	L	0		0	0	ļ .	Yes
SECURITIES ANALYSIS														Yes:
TI-WRITER (WORD PROCESSING)3	R	<u>. </u>		<u> </u>			Я	R	Ř		R	R	<u> </u>	Yes

*(*!

¥I ¥I

- The required items for speech are a synthesizer and one of the following: Extended Basic,
 TE II or Speech Editor, a module with built-in speech (Early Reading, for example) or PASCAL.
- ** Assembly Language also may be developed by using a P-Code Card, Memory Expansion Card, Peripheral Expansion System, Disk Drive and Controller and P-System Editor/Filer/Utility and Assembler/Linker Disks. You can run, but not write Assembly Language from Extended Basic,
- *** To run P-Code (PASCAL) Programs, the following is required: Peripheral Expansion System, Memory Expansion Card, P-Code Card and hardware appropriate for the software (Disk Drive and Controller for disks, cassette player and cables for tape, nothing else for cartridge). To write P-Code programs, the following is required: Peripheral Expansion System, Memory Expansion Card, P-Code Card, Disk Controller, at least one Disk Drive (two drives are better) and the P-System Editor/Filer/Utility and PASCAL Complier Disks.
- **** To run TI-PILOT lessons, the following is required: Peripheral Expansion System, Memory Expansion Card, P-Code Card, Disk Controller, at least one Disk Drive and the TI-PILOT Disk. The same items are required to write TI-PILOT lessons. The P-System Editor/Filer/Utility Disk also is required for writing lessons.

Stand-alone peripherals may be substituted for any function which requires the Peripheral Expansion System and one or more cards.

- † TI-BASIC can be run with Memory Expansion only when the Mini-Memory Cartridge is used.
- TI LOGO II (available late first quarter 1983) can use the RS-232 Interface Card and printer.
 TI LOGO can use only the II Thermal Printer.

[‡] Regents of the University of California

¹ A limited number may require use of cartridge and disk or cassette.

² Some require the Speech Synthesizer.

³ Available first quarter 1983 (1Q83)

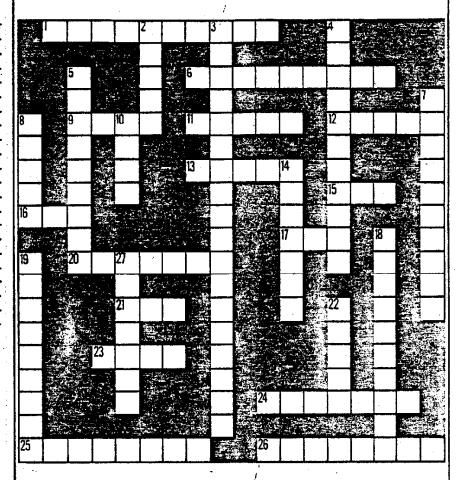
WISH LIST

*************** ***UQWVJRTVSEITILITUIEV*** *LOKLEOBFFSRE99VHZMSD* *CPZEJTBWAAIPAFQTQLIR* *OZUXZIEHADNMODULESSA* *LDBYVNXCSVJZYYMVCYEO* *WUPBIOPTSEVTEBVHOAHB* *VCBMSMAGENCQEMITKQTY* ***YQNOZXNCMTXWLXHCSHNE*** *LLCDIXSOBUSMAHINFWYK* *BPZEWYIRLRBALSUBXRSQ* *DRRMYVO/EENZANPSIEVG* *DRAOSONCRSZBTPRUSDFO* *DVIIGVKOHMXLOHWCKRTO* *IPMVNRCMKGWE&DITCOGD* *SDERESAPGAGTSRNEICKS* *CGMUHSOMFSJAFDDFTEZP* *STOJRAZMSZRBCLOUSRMO* *IYRFRS232CARDSWLYFDU* *KOYBCJTTXHPARGSKONCS* *J@WKYPRINTERFJLCJKZE*

WISH LIST

RS232CARD **EXPANSION** BRAINS TIME MODULES . DISCS MEMORY DRIVES PRINTER MODEM **PROGRAMS** ASSEMBLER GRAPHX **XBASIC** SYNTHESISE COR/COMP MONITOR **ADVENTURES** UTILITIES SWODNIN JOYSTICKS RECORDER GOODSPOUSE KEYBOARD HV99ERS

TI-CROSSWORD



ACROSS

- 1. Addition to System
- 6. Translator which makes two different peripherals compatible
- 9. Children's language
- 11. Monster who chases Munchman
- 12. Sometimes causes input error
- 13. Language built/into TI-99/4A
- 15. Command to process program
- 16. Command to leave program
- 17. Mathematical term
- 20. Program documentation within actual program
- 21. Command to automatically number each line
- 23. Terminates program execution
- 24. Process of correction
- A danger when trying to find the Wumpus
- 26. The first aliens to attack in the game PARSEC

DOWN

- 2. New Educational Software
- 3. Temporary space
- 4. Number 1 of color code
- 5. Command to make objects disappear on screen
- 7. Cassette Access
- 8. Command to transfer control to a different routine
- 14. Basic computer unit
- 18. Command to change line numbers
- 19. Inexpensive storage
- 22. Command that gives hard copy
- 27. Type of screen



BOLUTIONS ELSEWHERE

PETE SMITH HV99

MINI-WP DISSECTED

This month we will continue on line 220 and CALL DL(L\$(),L), DL is short for delete line. We find SUB EL(L\$(),L) at line 620, and to continue the confusion I again called the array L# and the line counter L in this subprogramme also. The sceen is cleared and the subprogramme LI is called. which Was one □f subprogrammes dealt with last month. Next line 630 the subprogramme LC is called, another subprogramme we dealt with last month. As you can see writing programmes that are full of subprogrammes is dead easy, you just keep using them over and over again. But on to line 640 where the length of the line L#(D) is added to the VAL of L\$(0) and all the lines are moved back one. That is if D=24 then L\$(24) now equals what L\$(25) equals and L\$(25) equals what L\$(26) is etc., up to the last text line. Then as $L^{\pm}(L-1)$ is the same as $L^{\pm}(L)$, L=(L) is set to a null string and L is decremented by one, and the subend is performed back to the main programme and the return at line 220

Next line 230 and INS(L#(),L), INS is short for insert and SUB INS(L=(),L) is found at line 450, and this is the subprogramme used to insert a line inbetween the ather lines of text. Again screen is cleared and LI is called again, but this time with a slightly different message. LC was not called this time as a line is not being changed or deleted, and if you found you made a mistake by inserting the line in the wrong place it is easy to get rid of it by selecting Delete Line from the main menu. On to line 660 where the line counter L is incremented by one and all the text lines are moved up one place from D+1, next a dummy CALL KEY 5 is called to turn on the full key board again, and then the programme waits for you to LINPUT the new line of taxt La(D), then the length of La(D) is subtracted from the VAL of L\$(0) to tree the omount of free available for text up to date. After that the SUBEND is performed back to the main Dragnamme and the return at line 230.

Only a short article this month as the next months subprogramme is a little more complicated and is bossibly an article in it self. Again if any of you have any questions don't be afraid to ask.

Brian R.

JNR. SOFTWARE REVIEW NO. 1

JUNIOR SOFTWARE REVIEW

AUSSIE GUESS.

This is quite an enjoyable program. It has attractive graphics with the unique ability of being able to write your own questions. It is a good one for the kids and is an excellent program The writer must have been a real genius. (CRAWL-CRAWL).

Rodney Gainsford

JNR. SOFTWARE REVIEW NO. 2

BATTLESHIP

"Battleship" is playled on a 20 \times 20 grid, based on the board game. The player and the computer place their ships on the grid unknown to each other.

The object of the game is to shoot and sink the computers ships. You have to guess where the ships are and enter those co-ordinates. After you have your tur the computer has its turn.

The player and the computer have -

- 1 CARRIER
- 4 SUBMARINES
- 4 FRIGATES

Also recorded on the screen is how many shots you have, the computers score and your score. When you have a shot at the computer it is recorded on the same grid as your ships.

Battleship is a great game to play and is available from the club library.

JASON HILES

32K MEMORY THE GARY JONES WAY

In the writing of this article thanks and acknowledgement of 99/4A dedication and expertise the members of TIUP Australia and in particular Bernie Elsner and the people behind the development of the 32K Matchbox Memory Expansion. provided the technical guidelines and background for this particular the same task implimentation of 32K memory expansion the cheap way. The reasons for the change from matchbox type construction to this on-board PC board method threefold :-

- 1) This type of construction reduces need for professional-type ability, particularly when soldering directly to chip legs. It takes this project to almost kit type construction. (However it is not recommended that electronics beginners carry out this modification).
- 2) This method allows for an isolation point for fault finding purposes, removal of memory or future modification by utilization of the 34 way ribbon cable connector between the machine and the memory expansion board.
- 3) It is still relatively cheap to upgrade to 32K at around \$A50. The PC board does not significantly increase construction price.

As mentioned before, this type of construction is almost kit experience however some electronics construction is needed. must say at this point that this is not all that difficult a job, but it is <u>very</u> tedious and attention to detail and constant checking is must. Speed does not enter into this project! It is also important to realise that any risk involved must be the responsibility of machine owner.

Now to the task at hand. physically object is to fit in a limited space, a memory expansion PC board with a 34 ribbon cable attached and wire this cable of the GROM port ontó the back (where the modules plug in) and to mother board where required. There are several things to be aware of before you start:-

You are working with IC's and these clever little devices have enemies, the main ones being static electricity and heat. It is therefore essential that you provide protection against both in the form of an anti-static mat with suitable earthing and the correct soldering iron with a clean tip.

When soldering to ICs in particular, remember to get in and out quickly but properly - check for dry joints.

It is also important to prepare and terminate wiring correctly. This means twisting wire strands together so that there are no loose strands and pre-soldering before the actual joint is made.

Now we begin the actual fitting.

- Set up your work area, earthing, tools required and get yourself comfortable.
- 2) Dismantle your machine carefully, taking note of screw positions and orientation of the boards. Be very careful when the mother board (the one in the metalwork) is lifted out as it connects to the keyboard via a pin connector underneath the board and is susceptible to breakage if not carefully removed. Once the mother board is removed, take note of the direction that the securing

20 he ir

he

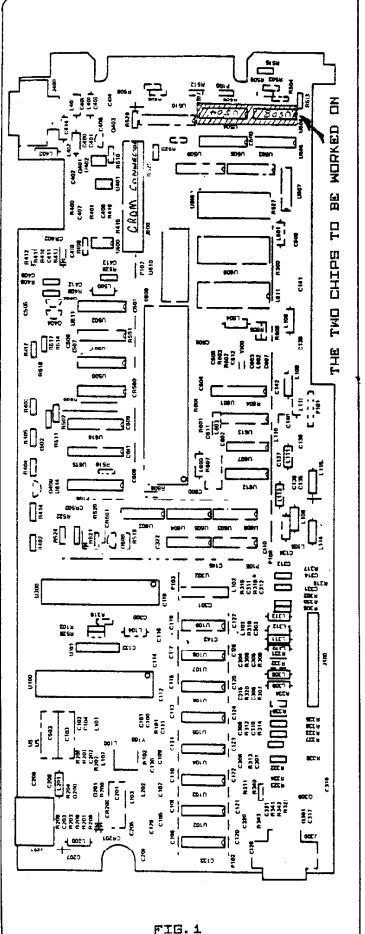
ĊO

pot You

rou ere ter

how ers eve is

lay lub



screws pass through the metalwork and board and remove them and the component side metalwork.

3) Identify the work area on the mother board referring to Fig.1 & 2 of the attached drawings and in particular chips U504 and U508 and the legs of those ICs that connections are to be made to. These are the chip enable lines and earth.

4) Remove GROM connector

- 5) Prepare your 34 way ribbon cable to a length of 275mm (11") and identify cable No.1 by the connector marker arrow.
- 6) To start wiring refer to Fig.2c and start laying these chip enable lines ta at a respective IC legs, one time, separating each wire from the the ribbon and stripping length back to the full connector. Remember when soldering to the chips in/out quickly but properly.
- 7) After all wires are connected, route the 5 wires to the outlet area at the rear of the GROM port around CR402, directing them as a whole (taped together) to the left and rear of the GROM port. Protect the edges of the topside metalwork in the outlet area and refit the topside metalwork after checking the wiring again.
- 8) Clean and refit GROM connector.
- 9) Referring to Fig.3, examine the location of the extension board with respect to the GROM port, taking particular notice of the physical limitation of the computer casing in that area.
- 10) Check the extension board for foreign matter, shorts and dry joints and if all is OK, apply double sided tape to the connection side full length and full width of the board.
- fit the 34 way ribbon cable to the connector on the extension board and after once more checking the positioning on

the prodov 12) GRC out por imp will tha

wir will thanker rea the por pos

ext

Nea

13)
wir
GRO
nea
num
hav
ythe

in service the the

por 14) 1de bre

acr

ext

15)
show in the year of the last of the la

wire chec last done spar the metalwork, remove the tape protection and stick the board down.

- 12) Now refer to Fig. 2a of the GROM port wiring and check it out physically with the GROM port. At this stage it i s important to realise that the wiring from board to GROM port will be restricted in the area that it lays in, and must be kept as close as possible to the rear of the GROM port, and, the area between the board and port must lay as flat as possible, with all wiring crossovers as close to the extension board as possible. Neatness in wiring practice certainly helps here!
- Refer to Fig. 2b and start wiring the bottom row of port from furthest GROM to nearest. This is ODDS the The wires will numbered row. have to be laid and trimmed as you go. Check each wire from the cable and its connection point (the pin) as you go, and in particular the first second wires laid, as any error here will flow through the whole laying sequence. Continue with the whole bottom row and when the last one is laid check the whole lot once more then tidy up wire laying area and tape the wires in a single layer across the metalwork between the extension board and the GROM port.
- 14) At this stage it may be an idea to stop, have a coffee break and relax for a while.

9

М

n M

ţе

ď

ηđ

Κ,

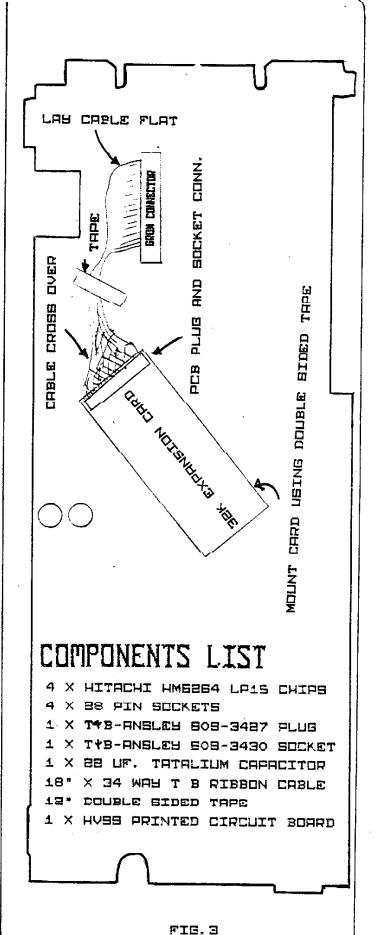
he th

Цe

he

te

15) Once back on the job you should have a fair idea of the task, and that is next the wiring of the top row of the GROM port - this is the EVENS Again check and recheck row. the first wire laid and when the wire, laying drop it vertically from the pin between two of the lower row pins, and then follow the same path as the other wiring to the extension board wiring. Continue wire by wire as before, continually checking until you reach the last pin - pin 2. When you have done this there should be 2 spare wires from the ribbon



REAR VIEW OF SROW FORT EXTENDER

(33)	34	3	(1)	(3)	9	(1)	0	(b)	(8)	(0)	0	(B)	90	<u> </u>	4	4	3	€ WIRE	NUMBERS
(34))	36	30	3.9	04	(1)							(2)			<u>-</u>	-	L PIN	NUMBERS
35	33	31	29	27	25	23	21	19	17	15	13	1	9	7	5	3		<u> </u>	

FIG. 29

WIRING SECLENCE

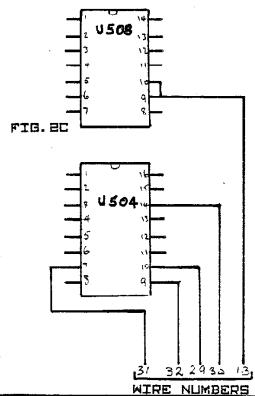
and the second of the second

WIRE	DIN	
	P 1/1	
34	35	
<u> </u>	45	
	14	
<u> </u>	19	
22	17 3	
24	DWER ROW	
26	13	
27		
25	9 💥	
23		
21	253 23 29 29 29 29 29 29 29 29 29 29 29 29 29	
17) 	FIG. 9B
33	36 7	
	37	
-11	30	
15 ,	44	
	7.6	
-4	2 4	
-	22 20 -9 -6 -10 -10 -10	
<u>&</u>	<u> </u>	
10	<u> </u>	
12	14 17	
14	12	
34 18 1 5 22 24 26 27 25 23 21 19 33 31 15 9 4 7 6 8 10 12 14 16 20 28	3208 2642 094 42 98 12 4	
20	8	
2 2	2.1	

P1-3-17-02	<u>1 1⊤14</u>
13 .	U508-9
32	U 50·4 - 9
29	U 504 - 10
<u>3</u> 0	U504-14
3	4504-7

DTN

WITOF .



cable. One of these has +5 Volts on it, so tape them individually and lay them along the metalwork near the extension board front edge and tape them to the metalwork. Tidy up the GROM port area wiring as before and tape in another flat layer on top of the bottom row wiring previously done.

- 16) Have a good look at the whole job now as you're getting close to completion. Check that the ribbon cable is plugged into the extension board properly and that the GROM connector is also plugged in properly.
- 17) Re-assemble your machine carefully and if everything is correctly located there should be no problem here.

18) Now to the moment of truth connect up your machine and go to Extended Basic. You can take another breath when the READY comes up. Once you have established the READY, type in SIZE and presto heaps of memory and many applications open up to you with little this mighty machine, particularly with the assistance of Will McGovern's tape loader program.

Many thanks to Gary Jones for his work on this particular implementation of this project, and we at Newcastle wish all that make use of it the very best.

4A - 4EVER

Albert Anderson/Gary Jones

"TEMESBOMB"

TIMESBOMB was a group project completed by the members of Garry Jones's Easic class. The quality of the completed program is a reflection upon Garry's teaching ability as well as the enthusiasm of his pupils.

TIMESBOMB is a very entertaining program which will have your children learning their times tables very quickly as they try to complete the alloted number of questions before the fuse burns out. The program is very flexible and allows you to adjust the rate at which the fuse burns.

```
100 REM ## SET ARRAY LIMIT
                                     370 CALL CHAR(97, "COCOCOCOCO
                                     COCOCO")
##
                                     380 CALL CHAR(98, "0303030303
110 DIM N(12)
                                     030303")
120 CALL CLEAR
130 CALL SCREEN(15)
                                     390 CALL CHAR(99, "FFFF000000
                                     000000*)
                                     400 CALL CHAR(100, "000000000
140 REM ## PRINT HEADER ##
150 P=14
                                     000FFFF")
                                     410 CALL CHAR(104, "3C3C3C3C3
160 W="WELCOME TO"
170 R=10
                                     C3C3C3C*)
180 GOSUB 270
                                     420 CALL CHAR(112, "183830303
190 W$="TIMESBOMB"
                                     81C1C18")
                                     430 CALL CHAR(113, "002072079
200 R=12
210 GOSUB 270
                                     D180000")
220 W#="A HV99'ERS GROUP PRO
                                     440 CALL CHAR(114, "0307063E3
JECT"
                                     C4040C0")
230 R=14
                                     450 CALL CHAR(115, *COE07C1E0
240 GOSUB 270
                                     E040303")
250 GOTO 310
                                     460 GOSUB 500
                                     470 GOSUB 610
260 REM ## PRINTING SUBROUT
                                     480 GOTO 710
INE ##
                                     490 REM ## BOMB SUBROUTINE
270 FOR W=1 TO LEN(W$)
280 CALL HCHAR(R,P-LEN(W$)/2
                                     ##
+W, ASC (SEG $ (W$, W, 1)))
                                     500 CALL VCHAR(19,26,98,6)
290 NEXT W
                                     510 CALL HCHAR (24,27,100,3)
300 RETURN
                                     520 CALL VCHAR(19,30,97,6)
310 P=16
                                     530 CALL HCHAR(19,27,99,3)
                                     540 CALL VCHAR(1,28,104,18)
320 REM ## ASSIGNING GRAPHI
                                     550 CALL HCHAR (20, 28, 66)
CS COLOURS ##
                                     560 CALL HCHAR(21,28,79)
                                     570 CALL HCHAR (22, 28, 77)
330 CALL COLOR(9,5,1)
340 CALL COLOR(10,9,1)
                                     580 CALL HCHAR (23, 28, 66)
350 CALL COLOR(11,2,1)
                                     590 RETURN
                                      600 REM ## WICK SUBROUTINE
360 REM ## DEFINING GRAPHIC
S CHARACTERS ##
                                      ##
```

1070 REM ## ACCEPT PLAYERS 610 FOR A=1 TO 18 620 CALL VCHAR(A, 28, 112) NAME ## 1080 PRINT " ENTER YOUR FIRS 430 FOR D=1 TO 50 T NAME": : 640 NEXT D 1090 INPUT NAMES 650 NEXT A 660 CALL SOUND (1000, -7,2) 1100 REM ## TEST PLAYERS NA 670 FOR DELAY=1 TO 300 ME ## 680 NEXT DELAY 1110 IF LEN(NAME\$)>20 THEN 1 **690 RETURN** 210 1120 IF NAMES="" THEN 1250 700 REM ## PLAYING INSTRUCT 1130 FOR A=1 TO LEN(NAME\$) IONS ## 710 CALL CLEAR 1140 B\$=SEG\$(NAME\$,A,1) 1150 IF B#=" " THEN 1230 720 PRINT " DO YOU WANT INST 1160 IF B\$>"Z" THEN 1270 RUCTIONS": : 1170 IF (B\$>"/") *(B\$<":") THE 730 PRINT " PRESS Y OR N" N 1290 740 CALL KEY(0,K,S) 750 IF S=0 THEN 740 1180 NEXT A 760 IF K<>89 THEN 1050 1190 REM ## DISPLAY ERROR M 770 CALL CLEAR 780 W=="TIMESBOMB HAS BEEN D ESSAGES ## 1200 GOTO 1310 ESIGNED" 1210 PRINT " YOUR NAME IS TO 790 R=3 800 GOSUB 270 O LONG": : 810 W=="TO MAKE LEARNING YOU 1220 GOTO 1080 1230 PRINT : : " FIRST NAME O R* NLY PLEASE ": : 820 R=5 1240 GOTO 1080 830 GOSUB 270 1250 PRINT : : " YOU MUST ENT 840 W#="TIMES TABLES MORE FU ER A NAME": : N. " 850 R=7 1260 GOTO 1080 1270 PRINT : : * PLEASE PRESS 840 GOSUB 270 ALPHA LOCK ": : 870 WS="THE SPEED CHOSEN DET 1280 GOTO 1080 ERMINES" 1270 PRINT : : PLEASE DO NO 880 R-9 T USE NUMBERS ":: 890 GOSUE 270 900 W#="HOW QUICK THE WICK B 1300 GOTO 1080 URNS." 1310 CALL CLEAR 1320 CALL SCREEN(4) 910 R=11 920 GOSUB 270 1330 REM ## GREET PLAYER ## 930 W#="YOU HAVE TO ENTER AL L THE" 940 R=13 1340 W\$="HELLO "&NAME\$ 950 GOSUB 270 1350 R=12 940 W#="QUESTIONS CORRECTLY 1360 GOSUB 270 1370 FOR DELAY=1 TO 300 BEFORE" 970 R=15 1380 NEXT DELAY 1390 CALL CLEAR 980 GOSUB 270 990 W#= "THE BOMB EXPLODES." 1400 J=0 1000 R=17 1410 REM ## ACCEPT TABLE NU 1010 GOSUB 270 1020 PRINT " PRESS ANY KEY T MBER AND TEST IT ## 1420 PRINT * ENTER TIMES TAB O CONTINUE " LE NUMBER": : 1030 CALL KEY(0,K,S) 1040 IF S=0 THEN 1030 1430 CALL SCREEN(12) 1440 PRINT " IT MUST BE BETW 1050 CALL CLEAR EEN 1 AND 12": : 1060 CALL SCREEN(12) 1450 INPUT T

```
1460 IF (T<1)+(T>12)THEN 144
                                    1890 FOR D=1 TO 2
                                    1900 CALL HCHAR(A,8+D,ASC(SE
                                    G$(N$,D,1)))
1470 REM ## ACCEPT SPEED AN
                                    1910 NEXT D
                                    1920 CALL HCHAR(A, 12, 61)
D TEST IT ##
1480 PRINT : " ENTER SPEED BE
                                    1930 NEXT A
TWEEN 1 AND 9": :
                                    1940 I=0
                                    1950 CALL SCREEN(12)
1490 PRINT * 1 IS SLOW - 9 I
S FAST": :
1500 INPUT SP
                                    1960 REM ## ACCEPT ANSWERS
1510 IF (SP(1)+(SP)9)THEN 14
                                    ##
                                    1970 ROW=0
90
                                    1980 FOR A=2 TO 24 STEP 2
1520 CALL CLEAR
1530 CALL SCREEN(4)
                                    1990 I=I+1
1540 PRINT "INITIALIZING...P
LEASE WAIT"
                                    2000 REM ## CALCULATE CORRE
                                    CT ANSWER ##
                                    2010 R1=T*N(I)
1550 REM ## LOAD 1 TO 12 IN
                                    2020 GOTO 2050
TO ARRAY IN RANDOM ORDER
1560 REM
             AND TEST THAT T
                                    2030 REM ## BEEP AND CLEAR
HERE ARE NO DOUBLE UPS ##
                                    IF ANSWER INCORRECT ##
1570 RANDOMIZE
                                    2040 CALL SOUND(100,-3,1)
1580 FOR I=1 TO 12
                                    2050 CGL=16
1590 N(I)=INT(13*RND)
                                    2060 ANS$=""
1600 FOR C=1 TO I
                                    2070 CALL HCHAR(A, COL-3, 32, 4
1610 IF N(I)=N(I-C)THEN 1590
1420 NEXT C
1630 NEXT I
                                    2080 REM ## PUT QUESTION MA
1640 T=""
                                    RK AT ANSWER POSITION ##
1650 IF T>9 THEN 1680
                                    2090 CALL HCHAR(A, 16, 63, 1)
1660 REM ## ADD SPACE TO NU
                                    2100 FOR KEY=1 TO 4
MBERS LESS THAN 10 ##
                                    2110 CALL KEY(0,K,S)
1670 T#=" "
                                    2120 IF K=13 THEN 2230
1680 T$=T$&STR$(T)
                                    2130 REM ## INCREMENT TIMIN
1690 CALL CLEAR
                                    G ##
1700 SP$=STR$(SP)
                                    2140 IF S(1 THEN 2550
1710 P=21
1720 W#=" SPEED IS "&SP#
                                    2150 REM ## PRINT ANSWER ON
1730 R=1
1740 GOSUB 270
                                     SCREEN ##
1750 P=16
                                    2160 ANS=ANS=&CHR=(K)
                                    2170 COL=COL-1
1760 GOSUB 500
                                    2180 FOR B=1 TO LEN(ANS$)
1770 I=0
                                    2190 CALL HCHAR(A, COL+B, ASC(
1780 REM ## PRINT TABLE ON
                                    SEG#(ANS#, B, 1)))
                                    2200 NEXT B
SCREEN ##
1790 FOR A=2 TO 24 STEP 2
                                    2210 NEXT KEY
1800 I=I+1
                                    2220 REM ## COMPARE ANSWERS
1810 FOR B=1 TO 2
1820 CALL HCHAR(A, 3+B, ASC(SE
                                      ##
                                     2230 IF STR#(R1) < >ANS# THEN
G$(T$,B,1)))
1830 NEXT B
                                     2040
                                    2240 NEXT A
1840 CALL HCHAR(A,7,88)
                                    2250 CALL SCREEN(4)
1850 Ns=""
1860 IF N(I)>9 THEN 1880
                                    2260 FOR DELAY=1 TO 400
1870 N#=" "
                                    2270 NEXT DELAY
1880 N#=N#&STR#(N(I))
                                    2280 CALL CLEAR
```

2290 REM ## CONGRATULATE PL AYER ## 2300 W=="WELL DONE "&NAME\$ 2310 R=13 2320 GOSUB 270 2330 FOR DELAY=1 TO 400 2340 NEXT DELAY 2350 REM ## REQUEST REPEAT 2360 PRINT " WOULD YOU LIKE TO TRY AGAIN": : 2370 CALL SCREEN(12) 2380 PRINT " PRESS 'Y' OR 'N 2390 CALL KEY(0,K,S) 2400 IF S=0 THEN 2390 2410 IF K=89 THEN 1390 2420 CALL CLEAR 2430 CALL SCREEN(4) 2440 REM ## SAY GOODBYE ## 2450 W=="PLAY AGAIN SOON" 2460 R=11 2470 GOSUB 270 2480 W=="BYE "&NAME= 2490 R=13 2500 GOSUB 270 2510 FOR DELAY=1 TO 400 2520 NEXT DELAY 2530 END 2540 REM ## TIMING SUBPROGR AM ## 2550 J=J+1 2560 IF J<>INT(100/SP)THEN 2 110 2570 ROW=ROW+1 2580 REM ## END OF TIME ## 2590 IF ROW=19 THEN 2640 2600 CALL VCHAR (ROW, 28, 112) 2610 J=0 2629 GOTO 2110 2630 REM ## EXPLOSION SUBR **DUTINE ##** 2640 CALL VCHAR(19,26,112,6) 2650 CALL VCHAR(19,30,112,6) 2660 CALL HCHAR(19,27,113,3) 2670 CALL HCHAR(24,27,113,3) 2680 FOR A=30 TO 1 STEP -1 2690 CALL SOUND (-400, -7, A)

2720 CALL HCHAR(J,K,L,M) 2730 NEXT A 2740 RESTORE 2750 CALL VCHAR(1,28,32,18) 2760 REM ## MESSAGE TO PLAY ER IF TABLE NOT COMPLETED IN TIME ## 2770 CALL CLEAR 2780 W#="BAD LUCK "&NAME# 2790 R=11 2800 GOSUB 270 2810 W#="THE SPEED YOU SELEC" TED" 2820 R=13 2830 GOSUB 270 2840 W\$= "WAS TOO FAST" 2850 R=15 2860 GOSUB 270 2870 FOR DELAY=1 TO 100 2880 NEXT DELAY 2890 GOTO 2360 2900 REM ## DATA FOR EXPLOS ION ## 2910 DATA 19,26,115,1,11,19, 30,114,1,7,21,26,114,1,7,21, 30,115,1,7,23,26,115,1,11,23 ,30,114,1,11 2920 DATA 18,25,113,1,11,18, 31,113,1,9,19,26,32,5,9,21,2

5,113,1,9,21,31,113,1,12,21,

2930 DATA 23,25,113,1,12,23,

31,113,1,10,23,26,32,5,10,20

,26,114,1,11,20,30,115,1,7,2

2940 DATA 22,30,114,1,7,24,2

6,114,1,11,24,30,115,1,11,17

,24,114,1,11,17,32,115,1,9,2

2950 DATA 21,32,114,1,9,24,2 4,114,1,12,24,32,115,1,12,20

,26,32,5,12,22,26,32,5,10,24

26,32,5,12

2,26,115,1,7

1,24,115,1,9

,26,32,5,10

2700 READ J,K,L,M,Q

2710 CALL SCREEN(Q)



ENTOMOLOGY EORNER FROM FUNIELVEB FARM

That time is here once again It has been another busy month or so fighting bugs the size of funnelwebs that seem to lurk in even the best intentioned of. program modifications. Somehow or other each step forward always comes with couple backwards. It always happens that when you start to push the limits of what a machine can do, that bugs turn up. The latest FUNLWRITER, not yet οf released for general consumption, is pushing every corner of the machine to the limits and turning up some very strange behaviour indeed. As yet I don't know whether it's just an error in my programming or a quirk or worse of the machine. Either is possible with the former more likely by far, but you never I think I mentioned in the first of this series that I found mysterious crashing of an Extended Basic program to be explainable only by a bug in CALL LINK.

The name of the HV79 group and Funnelweb Farm are spreading around the world now, and some interesting things are flowing back. Jingle Bells with a Dutch accent. One thing I am very happy to report is the receipt direct from Ottawa of Version II of the DM1000 disk manager program. The good news is that the fatal bugs of the first version have been fixed and the program can be used with confidence with no fear of trashing your files. This is great news as the program is a joy to use. The present version still has some problems. Richard tells me that the file functions only work on the first screen's worth of files. Since it does 20 per screen this is not a TI SSSD disks as we do here at FF. I expect this to be corrected as another version is in the works, which will support color choices and direct printing of DV/80 files.

have the source code DM1000 but haven't had time to even browse through it yet. When V3.0 of FUNLWRITER is ready to go we will be returning the favour in Ottawa is a group with a philosophy of operation very like that of HV99 and I am looking forward to a lot more cooperation with them in They have about members, which is not much more than twice our size. As a matter of empirical observation this general size range seems to be the most conducive to real activity contributions. Much larger and they seem to become a bunch 'gameplayers'.

At the time I wrote the warning in the last issue of the faults of first version the revised version was already on its way, but by surface mail. The bad news that came is that the original author DM1000 has indeed left the TI-99 fold. The good news is that he passed the rights to the program to the Ottawa group, and they continuing development. Strictly speaking the program is now issued "Freeware" by the Treasurer of the Ottawa TI-99/4a User group. Just think how much richer the world of software (and hardware mods) for the TI-99 would be if every User Group had its own pet project(s) of this quality.

Uttawa also sent a 2D graphics program in Forth by J-P. Morin. It looks good but we haven't yet gotten very far into it because it is on a double sided disk. More reports next time. Just in passing I have also heard from England that DM1000 has been a great boon for owners of Myarc disk controllers. These work fine but the disk manager supplied by Myarc requires either a Myarc expansion box or Myarc 128K card to run.

a joy to use. The present version still has some problems. Richard courtesy of Bernie Elsner of TIUP, and available from the Club functions only work on the first screen's worth of files. Since it does 20 per screen this is not a problem if you run only the standard language popularized by Bell Labs

along with Unix, though of much wider applicability. A close study of this will be my next computer project after FUNLWRITER V3.0 is wrapped up. It makes you realise what a dead hand TI's software policies laid on their flawed little gem of a machine. Now that TI has abandoned it, all sorts of interesting things are appearing. We can still only lament that the 99/8 never saw the light of day.

Another program which has come to us directly from its source is SUPERBUG II by Edgar Dohmann of the Johnson Space Center group (JUG) near Houston. This is a revision of the Superbugger program issued to user groups by TI when they orphaned the machine. It retains all the good features of Superbugger, such as disassembly, dumping to printer, software single stepping, while restoring the good parts of DEBUG that were omitted in Superbugger. The bugs in Superbugger such as inability to dump to disk and general fragility with keyboard entry have been fixed. The program comes in several versions for convenience in loading, including one which loads into an 8K RAM at >6000 if you have such a beastie. This is where cartridge ROM if any, resides and it comes up on the selection screen (1981 Version only) as a menu choice. We know from experience that this is a very useful option to have, and we have been using our own version of DEBUG prepared in this way for some time now. I might add that Will found that protection code in one heavily protected commercial offering included some to trash this area of memory just in case. Once again is 'Freeware' and will be available by the time you read this from the Club Library for the usual nominal fee.

Now that people are busily adding internal 32K console expansions we may see this extra 8K being added too. It can't be used all the time as many modules make use of this space already, most notably Extended Basic and Minimemory. You can tell by locking at a module if it does because metallic edge contacts are brought out on both sides of the module's Details have been circuit board. published in USA on do it yourself

projects to this end, company has produced a E/A module in this form in either kit or assembled form. Yes, E/A like TI-Writer one of those modules that contains only a single lousy GROM and nothing else. The extra memory can then be used just as memory or else, if you are not stuck with an 83 title screen console, to add new choices to the menu selection screen. All that it requires is that a standard ROM/GROM header as defined by TI in the technical manual be added to the programs loaded into the extra 8K. What TI did in the 83 consoles was to have the console ignore this ROM on its initial search for module programs. This meant that only programs with headers GROM. in exclusive to TI, would be loadable from the menu. We all know what a self-defeating policy that was.

While on the subject ٥f standard headers, I noticed that piece on the forthcoming Commodore Amiga in BYTE for Oct/85 remarked with approval and implicit contrast with existing small computer practice, that areas of memory were set aside for peripherals with ROM header code requirements carefully prescribed. This sort of thing of course is old hat to TI-99/4a users but apparently will be a big change for owners of things like Apples and Commodores which weren't designed by conservative engineers. From all the much: bally-hooed accounts Macintosh is a total disaster in this regard. The Amiga on paper at least, looks to be THE machine that will offer a genuine improvement for owners of TI-99s. It will mean that the present flood of new software, remarkable for an orphaned machine, will dry up as the writers go over to the more powerful machine. One reason the Amiga looks good is that Commodore neither designed it nor wrote its software. The only remaining hurdle is Commodore's abysmal record in quality control.

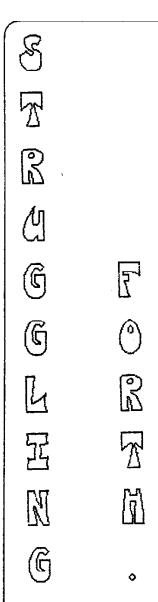
Anyway Merry Xmas to everyone. For the New Year I am thinking of starting a new series of articles on Assembly Language. As in the XB tutorials I will concentrate on the bigger issues in the framework of the machine as it exists, as there are already too many treatments which stay mired in nicky details and never get the big picture clear.

PI Si

b

u

0



FILE HANDLING.

This week an ATTEMPT!!! to write program which will accept ten items into an array as packed strings (strings with their count), open a Basic File on a non Forth disk, and then re-list it, together with an accompanying program in Extended Basic to access the same file and reprint the strings. This week I will stick to a fixed lengh internal format file with only one element per record so as not to complicate a subject I know very little about. get the feeling Tony McGovern should be writing this article as he really understands what he is doing, not just feeling his way like me!

Once you learn to open and close don't actually have to understand

interesting accessing existing files to examine the byte structure to sus out the codes indicating protection etc. Just for the heck of it, after I wrote this program, I wrote Basic Disk catalog in Forth, hence by trial and error I had to work out which bytes indicated protection etc. For the exercise I did it in forms- one just a straight record by record read and display, and the second a more sophisticated dump of the whole Basic Index file to an array and the VDP write from it to the screen which is much more elegant and impressively Whereas it is pointless writing yet another Basic disk manager program as there are some you beaut ones around such as the (hopefully almost) debugged DM1000 which is yours for the asking, it is actually useful to imbed a catalog word in the Forth dictionary when you are developing a Forth program which will be accessing Basic files, as it saves disorganised persons myself having to switch off re-enter a Basic environment ta catalog and then reboot. Next month I will discuss these listings in detail and also include a little screen dump of the printout for you.

Anyway down to business.

SETTING UP A PERIPHERAL ACCESS BLOCK.

A Peripheral Access Block is an area on memory set aside to contain information the computer will need to access peripheral devices, such as your disk or . printer. contains such information as the type of peripheral device eg DSK1 or DSK2, which record you wish to access etc.

Before like Arthur Dent you PANIC! at the complexity of the universe or in our case, the seeming complexity of the PAB, rest assured that if you consult your GUIDE, ie the FORTH MANUAL(somewhat less explicit than the HITCH-HIKERS GUIDE) all the necessary details are set out in a couple of quite clear examples. Arthur(who though remaining Like confused much οf the eventually reached his goal), you Basic files in Forth it is quite | what you are doing to be able to use

files, as long as you follow the format of their examples.

In reading data from a disk, or transferring it from an address in the bowels of your computer to the disk, it must first pass through the equivalent of a two temporary storage houses before being shunted

As the manual explains, the Device Service Routines built into the 99/4a data expect to perform transfers to/from VDP RAM. Our Forth uses CPU RAM as the site for data storage and manipulation hence the passage of data may look somthing like this:

DISK <-> VDPRAM <-> CPURAM <-> CPU BUFFER BUFFER ADR (ist half (2nd half way stop) way stop)

Addr: PAB-VBUF PAB-EUF TEST DATA

This actually is an over simplification as there is quite a lot more happening between taking information from the disk and it ending up in the VDPRAM buffer. For the inquisitive amongst you I am sure Tony McGovern would be only too happy to enlighten you in depth!

PAS-VEUF _____

Variable pointing to the VDF RAM buffer which serves as a temporary buffer when transfering data from pr to disk. We must designate this area from part of the "UNUSED" area on our memory map on page 3 Chapt 4. ensuring that we leave a long enough area to accommodate the record lengh we are pulling off/on disk.

PAB-BUF

This is our other half way temporary storage area. Like all other Forth words it is contained in CPU RAM in our dictionary and is allocated in the usual way for variables eg SCR# 41:

O VARIABLE BUFR 6 ALLOT. It does not have to be located within our definition of our file. It could even be an address we alternatively use for another purpose in our progam as long as we keep track of its usage.

EXAMPLE:

Say we have an array at TESTDATA, containing several three words. (Not the four expletives you are probably Emilling right now). For the moment let's not bother where they are or how they got there. To store the first word, say CAT to disk we must first pass it to our CPU RAM temporary half way house we have called BUFR using CMCVE:

or

bu

PA

Th

i e

a.d

fc

by

BU

P#

17

PA

No

st

Wh

FI

ar

WC

th

fu

Ðü

ab

if

THE

RE

S

ΕÝ

4

by ba

17 tĥ

ac

σŧ

do

WE

11

68

顶寨

.

рÞ

th

th

+4

×1

18

di

d∎

OP

--

Th

35

OB 50 ac

TESTDATA BUFR 3 CMOVE

If we examined BUFR using C@:

BUFR C@ . = C BUFR 1+ C@ . = A BUFR 2+ C@ . = T

From here, as long as we have set up our PAB (see Below) correctly, Forth will automatically transfer these bytes to the address WE designated in VDP RAM and on to the disk when we use the command REC-NO (Record number Incidently when writing definitions involving record tranfers and loops such as WRITE-REC on SCR# 42, when they don't work (not that I am suggesting for a moment that this would happen to you), examining the contents of our BUFR in this manner can help shed light on where we have gone astray.

LINKING THESE ADDRESSES TO A _____

FILENAME ____

REFER TO SCR# 42, LINE 6.

SCR 142

0 (FILE EXAMPLE - 230ct85)

1 : PAGE CLS 0 0 AT ;

0 2 AT ." NAME:" 5 2 : PROMPT

3 : GETARRAY PROMPT FILLBUF BETMANES ;

5 HEX

PABS @ A + BUFR 1700 FILE TESTFIL

7 DECIMAL

9 : SETFLE TESTFIL SET-PAR RLTV FXD INTRAL

6 REC-LEN F-D" DSK2.BBBB" ; 10

11

12 : WRITE-REC TESTDATA

13 10 0 BG DUP BUFR 4 CHOVE

14 I REC-NO 6 WRT 6 +

15 LOOP DROP :

I like to keep all my together and allocate them on the one screen, in this example SCR #41, but this does not matter. PABS A + :Starts our PAB A (HEX) bytes into this region. This starting point is our PAB-ADDR ie the peripheral access block Refer to Chapter 8 page 2 address.

for a full description of what each byte in the PAB describes. BUFR: Our half way CPU buffer

PAB-BUF

1700:Our half way VDP buffer PAB-VBUF

Now these three parameters are on the stack ie PAB-ADR, PAB-BUF, PAB-VBUF. When we execute:

FILE TESTFIL, these three parameters are permanantly associated with the word TESTFIL. We do not have to use the file straight away, but in future whenever we refer to TESTFIL our computer will associate above addresses with the name, even if another PAB as been used in the meantime.

REFER TO SCR#42, lines 9,10.

SETFLE (SETFILE)

Every time this Forth definition is encountered in a running program it will set the peripheral access block | SUFR: our temporary CPU PAB-BUF. at the address we originally designated (PABS A +). Into this in bytes 2/3 it will slot the VDP-RAM addrass we allocated ie 1700(PAB-VBUF), and it will zero all the other bytes in readiness to a accept information placed in it by other words ag REC-LEN atc. If we do not refer to this every time we want to operate on a particular file, ie leave it outside a definition such as in the Forth manual examples, it will work fine the first time, but if we or our program do somthing else such as use | fills TESTDATA with ASCII 32 blanks. the printer which sets up a PAB, It is nice to have clean memory. then if we again attempt to use our Sometimes its necessary as we may file the computer will lock up as it print out garbage if we don't, and will use the information from the PAB setup which will be different from what we originally designated.

OPENING AND CLOSING FILES.

file. These can be contained within definitions such as SCR#43, line 2. Remember you cannot open an open file or close a closed file or I/O errors will result.

EXAMINING THE PROGRAM LISTING.

SCREEN #41

SCR #41

```
0 ( FILE EXAMPLE - 230ct85)
 1 0 VARIABLE BUFR 6 ALLOT
 2 Q VARIABLE TESTDATA 60 ALLOT
 3 : AT
 4 : CONTINUE BEGIN PKEY 32 = UNTIL ;
5 : GET$
             TIB @ SWAP EXPECT 0 IN ! 32
              MORE HERE OVER OVER CO. DUP
              ROT C! 1+ 1 BO 1+ OVER OVER
              CE SWAP I + C! LOOP DROP
 8
             180P ;
10 : TITLE CLS 0 0 AT
              ." FORTH FILE HANDLING" ;
11
12 : FILLBUF TESTDATA 60 BLANKS (
13 : GETNAMES TESTBATA 14 4 BB 0 I AT I 3
             - . DUP 3 I AT 5 GET$ 6 +
14
```

TESTDATA

15

Space in our dictionary to accept 10 words we enter into an array.

CR LOOP BROP ;

GET\$

We examined last month. A handling word making a packed string [Adr, No char ---]

FILLBUF

at others keeping things clean makes examining of memory locations when we booboo visually easier!

GETNAMES

Does a loop 10 times (14 4) using this unusual loop index instead of There is no magic to this. As long | 10 0 to help position the cursor on as we set our PAB properly OPN will the screen at the right position, open the file (I got quite a buzz accepting up to 5 characters then seeing the drive light come on to incrementing TESTDATA by 6 + bytes access the file on my first|every time round(ie the address to attempts), and CLSE will close the which GET\$ deposits each new word)

th

5 p

FL

Dr.

56

WO

94

Th

DR

by

Pr

3¢

cu

92

```
SCREEN #42.
```

Most of this we have already examined.

WRITE-REC

Does a 10 0 loop moving data from TESTDATA TESTDATA 6 +

TESTDATA 12 + etc

to BUFR at the rate of 6 bytes (one entered word and it's preceding count) at a time. REC-NO then using the loop index I automatically shifts this data unaided by us to PAB-VBUF in VDP RAM and then onto disk. Our last unwanted address is DROPped to keep the stack clean.

SCREEN #43

```
SCR #43
 0 ( FILE EXAMPLE - 23DctB5)
 2 : WRITE-FILE SETFLE OPH WRITE-REC CLSE ;
 3
 4 : SAVE 0 23 AT
 5
          ." Press Space bar to save data"
 å
           CONTINUE 0 23 40 32 HCHAR 0 19
 7
          AT . * Opening File - Writing -*
 8
          WRITE-FILE ." File closed";
10 : READ-REC 10 0 BO 15 4 I + AT
11
                      ." Record:" I .
12
                      I REC-NO RB BUFR SWAP
13
                      TYPE
14
                   L08P ;
15
```

WRITE-FILE

Sets the correct PAB skeleton and file attributes with SETFLE as discussed above, OPNs the file, uses WRITE-REC to tranfer all 10 words to disk and the uses CLSE to close the basic file.

READ-REC

This is essentially the reverse of write record, tranfering data from Disk automatically to PAB-VBUF in VDP RAM and then to BUFR from where I have TYPEd it to screen. You will notice when running the program you get a blob before each word. This is the preceeding count which has not been EMITed hence it doesn't come out. If you change this

definition to DROP BUFR COUNT TYPE you will only get the word you originally typed in.

SCREEN #44 is self explanatory.

```
SCR 844
 0 ( FILE EXAMPLE - 230ct85)
 2 : REAB-FILE SETFLE OPN REAB-REC CLSE :
 4 : RETRIEVE 0 23 AT
              .* Press space bar to Relist*
 5
              CONTINUE 0 19 200 32 HCKAR
              15 2 AT ." RE-LISTING"
 7
              READ-FILE 15 15 AT
 a
 9
              ." FINISHED" CONTINUE :
10
11 : FILE-DENO TITLE GETARRAY SAVE
                RETRIEVE CONTINUE ;
13
15
```

SCREEN #45

:5

I put this in for easy reading of the source code for those of you (heaven forbid) who don't own a printer. You may have wondered why my listings are often fairly spaced out. Many reasons:
Legibility is a must, keeping things to 40 columns when using the 40 col editor means one does not have to keep zapping back and forth getting ones eyes twisted all the time. It

also leaves plenty of room if one

wants to add the stack diagram [before---after] at the end of each line. Since Forth is a compiled language and we can binary save the dictionary images, it is pointless writing illegible packed code, as no matter how many gaps on our source screen, no matter how

```
SCR #46
 0 ( FILE EXAMPLE - 230ct85)
 2 : FNENU PAGE ." FILE EXAMPLE-MENU" CR CR
                ." Choose option:"
                                     CR CR
 3
                ." 1. Run "
                                     CR
 5
                ." 2. List Code"
                                     C.P
                ." 3. Suit" ;
 7 : CHOICE 14 2 AT KEY DUP ENIT 48 -
            DUP ! = IF FILE-DENO
 8
                                     ELSE
 9
            BUP 2 = IF FLIST
                                     ELSE
 10
            DUP 3 = IF BROP PAGE QUIT ELSE
                  THEN THEN THEN DROP ;
11
12
13 : PROGRAM FMENU CHOICE MYSELF;
14
15
SCR #47
 0 ( FILE EXAMPLE - Load screen)
 2 41 LOAD 42 LOAD 43 LOAD 44 LOAD
 3 45 LOAD 46 LOAD
 10
п
12
13
14
```

many screens we space our code over, they still take the same amount of space in the dictionary!

FLIST

MESSAGE is used to sequentially print out the source code of each screen. Our loop indices are easily worked out with the definition I gave you last week:

Line=(scr#-4) +line no.

Eg start= 41-4 +0 = 592!
The definition of FLIST uses /MOD DROP to examine equal divisibility by 16. If the remainder is 0 the Press space Bar is exibited, the screen blanked using HCHAR and the cursor reset to the top of the screen (0 2 AT)

PROGRAM CHANGES FOR 1 DRIVE.

والراز وليتا للكراب والرازي والرازي والمنطور والراز ولياء والمتاهد للتاب المستعينيين والرازي المساويات والمعام

SCR #43 Line 9: F-D" DSK1.BBBB"

When running the program with one drive when the prompt comes up saying

"Press space Bar to save data", first insert the Basic File disk in Drive 1, then press the key. After re-listing re-insert your Forth disk before re-pressing the space bar.

RUNNING THE PROGRAM

After typing all the screens in type 47 LOAD PROGRAM and try it out. Mine works OK so hope yours does too.

I hope this has all helped a little. As I said, although it helps to understand what is happening during record transfers, especially when debugging, it is not 100% essential as by following the manual's formulae one can still be successful.

EXTENDED BASIC EQUIVALENT PROGRAM.

Just to prove it works and that E/B will understand your new file try the following program.

10 CALL CLEAR
20 OPEN #1:"DSK2.BBBB", RELATIVE, FIXED 6, INTERNAL
30 FOR A=1 TO 10
40 INPUT #1:A\$
50 PRINT A\$
60 NEXT A
70 CLOSE #1 :: END

Next Month a continuation of File handling with the examination of the two methods of accessing a Basic disks Index. This will introduce methods of printing to the screen using VDPWrites, and calculating offsets into arrays etc.

Any questions you may ring me on 22450(AH) or 436861(WK) if I'm not busy.

RICHARD TERRY DECEMBER 4TH 1985.

MIERRY CHIRISTIMAS

It's a well known fact that women like to have the last word, so as the Editors wife I've claimed the back page for myself.

Life in our house has certainly changed this year; April was the turning point. Not only did I give birth to our daughter Catherine, but Stephen (not to be outdone I suppose) gave birth to a magazine! Both baby and magazine are now seven months old and growing all the time. While I spend my days feeding baby and washing nappies, Stephen spends his days typing, pasting and pulling his hair out. occasionally pass each other on the staircase, he going up to do another article; me going down with a baby fresh from He goo-goos the baby, and I inspect the magazine. Then off we go on our separate "parenting" tasks. It all seemed so innocent at the time - "I'm going to be Editor for the club's new magazine." "That'll be nice, dear" says I; famous last words, thinking it would give him something to do in between painting the house, finishing the garage and digging our house out from the encroaching garden. Little did I realise then what a time consuming and demanding job it was.

Things normally get busy about two weeks before the magazine is due, and then it all starts - the long and frequent telephone calls, and day after day, and well into the night the strange men passing through my door. (I don't know what my neighbours think !!) These men all carry manilla envelopes and endless pages of printouts. They say a courtesy "hello" and say nice things about my baby, but I know they are all more interested in Stephens "baby". So 944 disappearing into the study for hours on end. As the deadline approaches the telephone calls and men increase. When at last its all over (for a week or two at least) he looks out the back door and comments how the pool has green and how the grass has grown.

It's a lot of hard work being a parent. I just wish we could swap "babies" occasionally because his at least doesn't cry in the night.

Denise Taylor HV99.

